Refine Search

Search Results -

Terms	Documents
L14 and (authentic$ same server)	1

Database:
US Pre-Grant Publication Full-Text Database
US Patents Full-Text Database
US OCR Full-Text Database
EPO Abstracts Database
JPO Abstracts Database
Derwent World Patents Index
IBM Technical Disclosure Bulletins

Search: L21

Refine Search

Recall Text | Clear | Interrupt

Search History

DATE: Saturday, August 14, 2004 Printable Copy Create Case

Set Name side by side	Query	Hit Count	Set Name result set
DB=EPAB,JPAB,DWPI,TDBD; THES=ASSIGNEE; PLUR=YES; OP=OR			
L21	L14 and (authentic$ same server)	1	L21
L20	L14 and (authentic$ with server)	0	L20
L19	L14 and (authentic$ same (cost$ or pric$))	0	L19
L18	L14 and (authentic$ same pric$)	0	L18
L17	L15 and (authentic$ same pric$)	0	L17
L16	L15 and (authentic$ with pric$)	0	L16
L15	L13 and @pd<=20001215	86	L15
L14	L13 and @ad<=20001215	113	L14
L13	"active x" or "active-x" or activex	198	L13
L12	L1	0	L12
DB=USPT; THES=ASSIGNEE; PLUR=YES; OP=OR			
L11	L2 and (authentic$ with pric$)	5	L11
L10	L6 and (authentic$ with pric$)	0	L10
L9	L8 and (server and client)	13	L9

L8	L7 and (authoriz$ or authentic$) and pric$	16	L8
L7	L5 and @pd<=20001215	29	L7
L6	L5 and @pd<=20001215	29	L6
L5	L4 and l2	48	L5
L4	705/26,27.ccls.	1168	L4
L3	705/26,27.ccls.l	1316862	L3
L2	L1 and @ad<=20001215	1480	L2
L1	"active x" or "active-x" or activex	1647	L1

END OF SEARCH HISTORY

Refine Search

Search Results -

Terms	Documents
L2 and L3	9

Database:
US Pre-Grant Publication Full-Text Database
US Patents Full-Text Database
US OCR Full-Text Database
EPO Abstracts Database
JPO Abstracts Database
Derwent World Patents Index
IBM Technical Disclosure Bulletins

Search:

Refine Search

Recall Text Clear Interrupt

Search History

DATE: Thursday, August 12, 2004 Printable Copy Create Case

Set Name side by side	Query	Hit Count	Set Name result set
DB=TDBD; PLUR=YES; OP=OR			
L4	l2 and l3	9	L4
L3	(1st or first or second or 2nd) adj server$	11	L3
L2	l1 and content or contents or data	26849	L2
L1	download$ or transfer$ or upload$ or sent or send or sends or sending	14835	L1

END OF SEARCH HISTORY

Hit List

Clear | Generate Collection | Print | Fwd Refs | Bkwd Refs | Generate OACS

Search Results - Record(s) 1 through 9 of 9 returned.

☐ 1. Document ID: NNRD42196

L4: Entry 1 of 9 File: TDBD May 1, 1999

TDB-ACC-NO: NNRD42196

DISCLOSURE TITLE: LAN Based Educational System

PUBLICATION-DATA:
Research Disclosure, May 1999, UK

VOLUME NUMBER: 42
ISSUE NUMBER: 421

Full | Title | Citation | Front | Review | Classification | Date | Reference | Sequences | Attachments | Claims | KWIC | Draw De

☐ 2. Document ID: NNRD41585

L4: Entry 2 of 9 File: TDBD Nov 1, 1998

TDB-ACC-NO: NNRD41585

DISCLOSURE TITLE: Linking a SAP* System to a MQSeries** Server

PUBLICATION-DATA:
Research Disclosure, November 1998, UK

VOLUME NUMBER: 41
ISSUE NUMBER: 415

☐ 3. Document ID: NNRD410116

L4: Entry 3 of 9 File: TDBD Jun 1, 1998

TDB-ACC-NO: NNRD410116

DISCLOSURE TITLE: Internet Based Secure Transactions Using Encrypting Applets and CGI-Scripts Independent of Browser or Server Capabilities

PUBLICATION-DATA:
Research Disclosure, June 1998, UK

VOLUME NUMBER: 41
ISSUE NUMBER: 410

☐ 4. Document ID: NN9801409

L4: Entry 4 of 9 File: TDBD Jan 1, 1998

TDB-ACC-NO: NN9801409

DISCLOSURE TITLE: Process for Transparently Locating and Running Applications on Servers

PUBLICATION-DATA:
IBM Technical Disclosure Bulletin, January 1998, US

VOLUME NUMBER: 41
ISSUE NUMBER: 1
PAGE NUMBER: 409 - 420

☐ 5. Document ID: NN951267

L4: Entry 5 of 9 File: TDBD Dec 1, 1995

TDB-ACC-NO: NN951267

DISCLOSURE TITLE: Method for Deadlock Prevention for Callbacks in a Distributed Environment

PUBLICATION-DATA:
IBM Technical Disclosure Bulletin, December 1995, US

VOLUME NUMBER: 38
ISSUE NUMBER: 12
PAGE NUMBER: 67 - 70

Full | Title | Citation | Front | Review | Classification | Date | Reference | Sequences | Attachments | Claims | KWIC | Draw De

---

☐ 6. Document ID: NN9502143

L4: Entry 6 of 9 File: TDBD Feb 1, 1995

TDB-ACC-NO: NN9502143

DISCLOSURE TITLE: Fine Granularity Locking to Support High Data Availability in a Client/Server Database Management System

PUBLICATION-DATA:
IBM Technical Disclosure Bulletin, February 1995, US

VOLUME NUMBER: 38
ISSUE NUMBER: 2
PAGE NUMBER: 143 - 146

Full | Title | Citation | Front | Review | Classification | Date | Reference | Sequences | Attachments | Claims | KWIC | Draw De

---

☐ 7. Document ID: NB9402221

L4: Entry 7 of 9 File: TDBD Feb 1, 1994

TDB-ACC-NO: NB9402221

DISCLOSURE TITLE: Dual Processor Design Using Gast Dual Port Scram

PUBLICATION-DATA:
IBM Technical Disclosure Bulletin, February 1994, US

VOLUME NUMBER: 37
ISSUE NUMBER: 2B
PAGE NUMBER: 221 - 224

☐ 8. Document ID: NN930194

L4: Entry 8 of 9 File: TDBD Jan 1, 1993

TDB-ACC-NO: NN930194

DISCLOSURE TITLE: Algorithmic Method for Distributing Clients Predictably and Evenly to Servers.

PUBLICATION-DATA:
IBM Technical Disclosure Bulletin, January 1993, US

VOLUME NUMBER: 36
ISSUE NUMBER: 1
PAGE NUMBER: 94 - 95

☐ 9. Document ID: NN9110451

L4: Entry 9 of 9 File: TDBD Oct 1, 1991

TDB-ACC-NO: NN9110451

DISCLOSURE TITLE: Model and Architecture for Diagnostic Requests in a Heterogeneous Distributed Environment.

PUBLICATION-DATA:
IBM Technical Disclosure Bulletin, October 1991, US

VOLUME NUMBER: 34
ISSUE NUMBER: 5
PAGE NUMBER: 451 - 455

Hit List

Clear | Generate Collection | Print | Fwd Refs | Bkwd Refs | Generate OACS

Search Results - Record(s) 1 through 10 of 13 returned.

☐ 1. Document ID: US 6134548 A

L9: Entry 1 of 13 File: USPT Oct 17, 2000

US-PAT-NO: 6134548
DOCUMENT-IDENTIFIER: US 6134548 A

TITLE: System, method and article of manufacture for advanced mobile bargain shopping

Full | Title | Citation | Front | Review | Classification | Date | Reference | Claims | KWMC | Draw D

☐ 2. Document ID: US 6119105 A

L9: Entry 2 of 13 File: USPT Sep 12, 2000

US-PAT-NO: 6119105
DOCUMENT-IDENTIFIER: US 6119105 A

TITLE: System, method and article of manufacture for initiation of software distribution from a point of certificate creation utilizing an extensible, flexible architecture

Full | Title | Citation | Front | Review | Classification | Date | Reference | Claims | KWMC | Draw D

☐ 3. Document ID: US 6073124 A

L9: Entry 3 of 13 File: USPT Jun 6, 2000

US-PAT-NO: 6073124
DOCUMENT-IDENTIFIER: US 6073124 A

TITLE: Method and system for securely incorporating electronic information into an online purchasing application

Full | Title | Citation | Front | Review | Classification | Date | Reference | Claims | KWMC | Draw D

☐ 4. Document ID: US 6072870 A

L9: Entry 4 of 13 File: USPT Jun 6, 2000

US-PAT-NO: 6072870

DOCUMENT-IDENTIFIER: US 6072870 A

TITLE: System, method and article of manufacture for a gateway payment architecture utilizing a multichannel, extensible, flexible architecture

Full | Title | Citation | Front | Review | Classification | Date | Reference | Claims | KWIC | Draw De

---

☐ 5. Document ID: US 6026379 A

L9: Entry 5 of 13 File: USPT Feb 15, 2000

US-PAT-NO: 6026379
DOCUMENT-IDENTIFIER: US 6026379 A

TITLE: System, method and article of manufacture for managing transactions in a high availability system

Full | Title | Citation | Front | Review | Classification | Date | Reference | Claims | KWIC | Draw De

---

☐ 6. Document ID: US 6012050 A

L9: Entry 6 of 13 File: USPT Jan 4, 2000

US-PAT-NO: 6012050
DOCUMENT-IDENTIFIER: US 6012050 A

TITLE: Multi-transaction service system

Full | Title | Citation | Front | Review | Classification | Date | Reference | Claims | KWIC | Draw De

---

☐ 7. Document ID: US 6002767 A

L9: Entry 7 of 13 File: USPT Dec 14, 1999

US-PAT-NO: 6002767
DOCUMENT-IDENTIFIER: US 6002767 A
**** See image for Certificate of Correction ****

TITLE: System, method and article of manufacture for a modular gateway server architecture

Full | Title | Citation | Front | Review | Classification | Date | Reference | Claims | KWIC | Draw De

---

☐ 8. Document ID: US 5987132 A

L9: Entry 8 of 13 File: USPT Nov 16, 1999

US-PAT-NO: 5987132
DOCUMENT-IDENTIFIER: US 5987132 A
**** See image for Certificate of Correction ****

TITLE: System, method and article of manufacture for conditionally accepting a payment method utilizing an extensible, flexible architecture

☑ 9. Document ID: US 5970475 A

L9: Entry 9 of 13 File: USPT Oct 19, 1999

US-PAT-NO: 5970475
DOCUMENT-IDENTIFIER: US 5970475 A

TITLE: Electronic procurement system and method for trading partners

☐ 10. Document ID: US 5943424 A

L9: Entry 10 of 13 File: USPT Aug 24, 1999

US-PAT-NO: 5943424
DOCUMENT-IDENTIFIER: US 5943424 A

TITLE: System, method and article of manufacture for processing a plurality of transactions from a single initiation point on a multichannel, extensible, flexible architecture

Clear | Generate Collection | Print | Fwd Refs | Bkwd Refs | Generate OACS

Terms	Documents
L8 and (server and client)	13

Display Format: TI Change Format

Previous Page Next Page Go to Doc#

Hit List

Clear | Generate Collection | Print | Fwd Refs | Bkwd Refs | Generate OACS

Search Results - Record(s) 1 through 1 of 1 returned.

☐ 1. Document ID: KR 2002030634 A

Using default format because multiple data bases are involved.

L21: Entry 1 of 1 File: DWPI Apr 25, 2002

DERWENT-ACC-NO: 2002-653141
DERWENT-WEEK: 200270
COPYRIGHT 2004 DERWENT INFORMATION LTD
TITLE: System for drawing type and digital instant lottery on internet

INVENTOR: PARK, J S

PRIORITY-DATA: 2000KR-0061665 (October 19, 2000)

PATENT-FAMILY:

PUB-NO	PUB-DATE	LANGUAGE	PAGES	MAIN-IPC
KR 2002030634 A	April 25, 2002		001	G06F017/60

INT-CL (IPC): G06 F 17/60

Full | Title | Citation | Front | Review | Classification | Date | Reference | Claims | KWIC | Draw De

Clear | Generate Collection | Print | Fwd Refs | Bkwd Refs | Generate OACS

Terms	Documents
L14 and (authentic$ same server)	1

Display Format: - Change Format

Previous Page Next Page Go to Doc#

End of Result Set

L21: Entry 1 of 1 File: DWPI Apr 25, 2002

DERWENT-ACC-NO: 2002-653141
DERWENT-WEEK: 200270

TITLE: System for drawing type and digital instant lottery on internet

Basic Abstract Text (2):
DETAILED DESCRIPTION - The system for a drawing type lottery comprises a database server(4), a web server(5), an ADO(ActiveX Data Object)(7) including an ODBC(Open Database Connectivity)(8), a client(9), and a housing lottery drawing database(6). The respective components exchange the data. A user is authenticated by using a member ID(Identifier) and a password. The authenticated user selects the numbers for a lottery on a lottery application page. The lottery numbers selected by the user are stored in the database server. A lucky number is selected by the data exchange between the web server and the housing lottery drawing database. The system for a digital instant lottery comprises a database server, a web server, an INET control(Microsoft Internet Transfer control), an ADO including an OLE(Object Linking and Embedding) database, a Winsock control, and a client.

PF Application Date (1):
20001019

L11: Entry 5 of 5 File: USPT Aug 21, 2001

US-PAT-NO: 6279112
DOCUMENT-IDENTIFIER: US 6279112 B1

TITLE: Controlled transfer of information in computer networks

DATE-ISSUED: August 21, 2001

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
O'Toole, Jr.; James W.	Cambridge	MA		
Gifford; David K.	Weston	MA		

ASSIGNEE-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
Open Market, Inc.	Cambridge	MA			02

APPL-NO: 08/ 741862 [PALM]
DATE FILED: October 29, 1996

INT-CL: [07] G06 F 11/30

US-CL-ISSUED: 713/201; 705/14
US-CL-CURRENT: 713/201; 705/14

FIELD-OF-SEARCH: 395/187.01, 395/188.01, 395/200.59, 380/21, 380/23, 380/24, 380/25, 713/154, 705/14, 705/51, 705/57, 705/59, 705/77, 705/80

PRIOR-ART-DISCLOSED:

U.S. PATENT DOCUMENTS

	PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
☐	5341293	August 1994	Vertelney et al.	
☐	5347632	September 1994	Filepp et al.	
☐	5450593	September 1995	Howell et al.	
☐	5455953	October 1995	Russell	395/739
☐	5490244	February 1996	Isensee et al.	
☐	5586260	December 1996	Hu	395/200.2

☐	5594921	January 1997	Pettus	
☐	5617565	April 1997	Augenbraun et al.	
☐	5673322	September 1997	Pepe et al.	
☐	5680452	October 1997	Shanton	
☐	5710918	January 1998	Lagarde et al.	
☐	5715314	February 1998	Payne et al.	380/24
☐	5717923	February 1998	Dedrick	395/613
☐	5724424	March 1998	Gifford	380/24
☐	5761648	June 1998	Golden et al.	705/14
☐	5809242	September 1998	Shaw et al.	395/200.47
☐	5838790	November 1998	McAuliffe et al.	380/4
☐	5948061	September 1999	Merriman et al.	709/219

FOREIGN PATENT DOCUMENTS

FOREIGN-PAT-NO	PUBN-DATE	COUNTRY	US-CL
WO 97/15885	May 1997	WO	

OTHER PUBLICATIONS

Open Market, Inc.; OM Express.TM. Information Area; http://www.openmarket.com/express; Jul. 29, 1996.

ART-UNIT: 275

PRIMARY-EXAMINER: Beausoliel, Jr.; Robert W.

ASSISTANT-EXAMINER: Baderman; Scott T.

ATTY-AGENT-FIRM: Fish & Richardson P.C.

ABSTRACT:

The present invention relates to techniques for controlling transfers of information in computer networks. One technique involves transmitting from a server computer to a client computer a document containing a channel object corresponding to a communication service, and storing an access ticket that indicates that a user of the client computer permits the information source computer to communicate with the user over a specified channel. Another technique involves transmitting smart digital offers based on information such as coupons and purchasing histories stored at the computer receiving the offer. Another technique involves transmitting from a server computer to a client computer a request for a user's personal profile information, and activating a client avatar that compares the request for personal profile information with a security profile of the user limiting access to personal profile information. Another technique involves transmitting from a server computer to a client computer a document containing an embedded link, activating the embedded link at the client computer and recording activation of the embedded link in a metering log.

L9: Entry 1 of 13 File: USPT Oct 17, 2000

DOCUMENT-IDENTIFIER: US 6134548 A
TITLE: System, method and article of manufacture for advanced mobile bargain shopping

Abstract Text (1):
A system is disclosed that facilitates web-based comparison shopping in conventional, physical, non-web retail environments. A wireless phone or similar hand-held wireless device with Internet Protocol capability is combined with a miniature barcode reader (installed either inside the phone or on a short cable) and utilized to obtain definitive product identification by, for example, scanning a Universal Product Code (UPC) bar code from a book or other product. The wireless device transmits the definitive product identifier to a service routine (running on a Web server), which converts it to (in the case of books) its International Standard Book Number or (in the case of other products) whatever identifier is appropriate. The service routine then queries the Web to find price, shipping and availability information on the product from various Web suppliers. This information is formatted and displayed on the hand-held device's screen. The user may then use the hand-held device to place an order interactively.

Application Filing Date (1):
19981119

DATE ISSUED (1):
20001017

Brief Summary Text (4):
The concept of agency, or the user of agents, is well established. An agent is a person authorized by another person, typically referred to as a principal, to act on behalf of the principal. In this manner the principal empowers the agent to perform any of the tasks that the principal is unwilling or unable to perform. For example, an insurance agent may handle all of the insurance requirements for a principal, or a talent agent may act on behalf of a performer to arrange concert dates.

Brief Summary Text (8):
According to a broad aspect of a preferred embodiment of the invention, A system is disclosed that facilitates web-based comparison shopping in conventional, physical, non-web retail environments. A wireless phone or similar hand-held wireless device with Internet Protocol capability is combined with a miniature barcode reader (installed either inside the phone or on a short cable) and utilized to obtain definitive product identification by, for example, scanning a Universal Product Code (UPC) bar code from a book or other product. The wireless device transmits the definitive product identifier to a service routine (running on a Web server), which converts it to (in the case of books) its International Standard Book Number or (in the case of other products) whatever identifier is appropriate. The service routine then queries the Web to find price, shipping and availability information on the product from various Web suppliers. This information is formatted and displayed on the hand-held device's screen. The user may then use the hand-held device to place an order interactively.

Drawing Description Text (27):
FIG. 25 is a block diagram of a back end server in accordance with a preferred embodiment; and

Detailed Description Text (41):
Thus, through the development of frameworks for solutions to various problems and programming tasks, significant reductions in the design and development effort for software can be achieved. A preferred embodiment of the invention utilizes HyperText Markup Language (HTML) to implement documents on the Internet together with a general-purpose secure communication protocol for a transport medium between the client and the Newco. HTTP or other protocols could be readily substituted for HTML without undue experimentation. Information on these products is available in T. Berners-Lee, D. Connoly, "RFC 1866: Hypertext Markup Language-2.0" (November 1995); and R. Fielding, H, Frystyk, T. Berners-Lee, J. Gettys and J. C. Mogul, "Hypertext Transfer Protocol--HTTP/1.1: HTTP Working Group Internet Draft" (May 2, 1996). HTML is a simple data format used to create hypertext documents that are portable from one platform to another. HTML documents are SGML documents with generic semantics that are appropriate for representing information from a wide range of domains. HTML has been in use by the World-Wide Web global information initiative since 1990. HTML is an application of ISO Standard 8879:1986 Information Processing Text and Office Systems; Standard Generalized Markup Language (SGML).

Detailed Description Text (42):
To date, Web development tools have been limited in their ability to create dynamic Web applications which span from client to server and interoperate with existing computing resources. Until recently, HTML has been the dominant technology used in development of Web-based solutions. However, HTML has proven to be inadequate in the following areas:

Detailed Description Text (48):
Sun Microsystem's Java language solves many of the client-side problems by:

Detailed Description Text (49):
Improving performance on the client side;

Detailed Description Text (52):
With Java, developers can create robust User Interface (UI) components. Custom "widgets" (e.g. real-time stock tickers, animated icons, etc.) can be created, and client-side performance is improved. Unlike HTML, Java supports the notion of client-side validation, off loading appropriate processing onto the client for improved performance. Dynamic, real-time Web pages can be created. Using the above-mentioned custom UI components, dynamic Web pages can also be created.

Detailed Description Text (53):
Sun's Java language has emerged as an industry-recognized language for "programming the Internet." Sun defines Java as: "a simple, object-oriented, distributed, interpreted, robust, secure, architecture-neutral, portable, high-performance, multithreaded, dynamic, buzzword-compliant, general-purpose programming language. Java supports programming for the Internet in the form of platform-independent Java applets." Java applets are small, specialized applications that comply with Sun's Java Application Programming Interface (API) allowing developers to add "interactive content" to Web documents (e.g. simple animations, page adornments, basic games, etc.). Applets execute within a Java-compatible browser (e.g. Netscape Navigator) by copying code from the server to client. From a language standpoint, Java's core feature set is based on C++. Sun's Java literature states that Java is basically "C++, with extensions from Objective C for more dynamic method resolution".

Detailed Description Text (54):
Another technology that provides similar function to JAVA is provided by Microsoft

and ActiveX Technologies, to give developers and Web designers wherewithal to build dynamic content for the Internet and personal computers. ActiveX includes tools for developing animation, 3-D virtual reality, video and other multimedia content. The tools use Internet standards, work on multiple platforms, and are being supported by over 100 companies. The group's building blocks are called ActiveX Controls, small, fast components that enable developers to embed parts of software in hypertext markup language (HTML) pages. ActiveX Controls work with a variety of programming languages including Microsoft Visual C++, Borland Delphi, Microsoft Visual Basic programming system and, in the future, Microsoft's development tool for Java, code named "Jakarta." ActiveX Technologies also includes ActiveX Server Framework, allowing developers to create server applications. One of ordinary skill in the art readily recognizes that ActiveX could be substituted for JAVA without undue experimentation to practice the invention.

Detailed Description Text (157):
Depending on what location the system identifies through pattern matching or alternatively depending on what location the user indicates as the meeting place, a system in accordance with a preferred embodiment suggests a plurality of fine restaurants whenever it detects the words lunch/dinner/breakfast. We can also use a site like company finder to confirm what we got is indeed a company name or if there is no company name that pattern matching can identify, we can use a company finder web site as a "dictionary" for us to determine whether certain capitalized words represent a company name. We can even display stock prices and breaking news for a company that we have identified.

Detailed Description Text (159):
FIG. 9 is a flow diagram that depicts the hardware and logical flow of control for a device and a software system designed to allow Web-based comparison shopping in conventional, physical, non-Web retail environments. A wireless phone or similar hand-held wireless device 920 with Internet Protocol capability is combined with a miniature barcode reader 910 (installed either inside the phone or on a short cable) and used to scan the Universal Product Code (UPC) bar code on a book or other product 900. The wireless device 920 transmits the bar code via an antennae 930 to the Pocket BargainFinder Service Module (running on a Web server) 940, which converts it to (in the case of books) its International Standard Book Number or (in the case of other products) whatever identifier is appropriate. The Service Module then contacts the appropriate third-party Web site(s) to find price, shipping and availability information on the product from various Web suppliers 950. This information is formatted and displayed on the hand-held device's screen. The IP wireless phone or other hand held device 920 utilizes a wireless modem such as a Ricochet SE Wireless Modem from Metricom. Utilizing this device, a user can hang out in a coffee shop with a portable computer perched on a rickety little table, with a latte sloshing dangerously close to the keyboard, and access the Internet at speeds rivaling direct connect via a telephone line.

Detailed Description Text (162):
Thus, utilizing the wireless modem, a user may utilize the web server software 940 to identify the right product 950 and then use an appropriate device's key(s) to select a supplier and place an order in accordance with a preferred embodiment. The BargainFinder Service Module then consummates the order with the appropriate third-party Web supplier 960.

Detailed Description Text (169):
FIG. 10A describes the Intention Value Network Architecture implementation for the World Wide Web. For simplification purposes, this diagram ignores the complexity pertaining to security, scalability and privacy. The customer can access the Intention Value Network with any Internet web browser 1010, such as Netscape Navigator or Microsoft Internet Explorer, running on a personal computer connected to the Internet or a Personal Digital Assistant with wireless capability. See FIG. 17 for a more detailed description of the multiple methods for accessing an

Intention Value Network. The customer accesses the Intention Value Network through the unique name or IP address associated with the Integrator's Web Server 1020. The Integrator creates the Intention Value Network using a combination of resources, such as the Intention Database 1030, the Content Database 1040, the Supplier Profile Database 1050, and the Customer Profile Database 1060.

Detailed Description Text (171):
The Supplier Profile Database 1050 contains information about the product and service providers integrated into the intention. The information contained in this database provides a link between the intention framework and the suppliers. It includes product lists, features and descriptions, and addresses of the suppliers' product web sites. The Customer Profile Database 1060 contains personal information about the customers, such as name, address, social security number and credit card information, personal preferences, behavioral information, history, and web site layout preferences. The Supplier's Web Server 1070 provides access to all of the supplier's databases necessary to provide information and transactional support to the customer.

Detailed Description Text (172):
The Product Information Database 1080 stores all product-related information, such as features, availability and pricing. The Product Order Database 1090 stores all customer orders. The interface to this database may be through an Enterprise Resource Planning application offered by SAP, Baan, Oracle or others, or it may be accessible directly through the Supplier's Web Server or application server. The Customer Information Database 1091 stores all of the customer information that the supplier needs to complete a transaction or maintain customer records.

Detailed Description Text (173):
FIG. 10B is a flowchart providing the logic utilized to create a web page within the Egocentric Interface. The environment assumes a web server and a web browser connected through a TCP/IP network, such as over the public Internet or a private Intranet. Possible web servers could include Microsoft Internet Information Server, Netscape Enterprise Server or Apache. Possible web browsers include Microsoft Internet Explorer or Netscape Navigator. The client (i.e. web browser) makes a request 1001 to the server (i.e. web server) for a particular web page. This is usually accomplished by a user clicking on a button or a link within a web page. The web server gets the layout and content preferences 1002 for that particular user, with the request to the database keyed off of a unique user id stored in the client (i.e. web browser) and the User profile database 1003. The web server then retrieves the content 1004 for the page that has been requested from the content database 1005. The relevant user-centric content, such as calendar, email, contact list, and task list items are then retrieved 1006. (See FIG. 11 for a more detailed description of this process.) The query to the database utilizes the user content preferences stored as part of the user profile in the User profile database 1003 to filter the content that is returned. The content that is returned is then formatted into a web page 1007 according to the layout preferences defined in the user profile. The web page is then returned to the client and displayed to the user 1008.

Detailed Description Text (174):
FIG. 11 describes the process of retrieving user-centric content to add to a web page. This process describes 1006 in FIG. 10B in a more detailed fashion. It assumes that the server already has obtained the user profile and the existing content that is going to be integrated into this page. The server parses 1110 the filtered content, looking for instances of events, contact names and email addresses. If any of these are found, they are tagged and stored in a temporary holding space. Then, the server tries to find any user-centric content 1120 stored in various databases. This involves matching the tagged items in the temporary storage space with calendar items 1130 in the Calendar Database 1140; email items 1115 in the Email Database 1114; contact items 1117 in the Contact Database 1168;

task list items 1119 in the Task List Database 1118; and news items 1121 in the News Database 1120. After retrieving any relevant user-centric content, it is compiled together and returned 1122.

Detailed Description Text (185):
FIG. 15 describes the process for generating the page that displays the agent's current statistics. When the user requests the agent statistics page 1510 with the client browser, the server retrieves the users' statistics 1520 from the users' profile database 1530. The server then performs the mathematical calculations necessary to create a normalized set of statistics 1540. The server then retrieves the formulas 1550 from the content database 1560 that will be used to calculate the user-centric statistics. Graphs are then generated 1570 using the generic formulas and that user's statistics. These graphs are inserted into a template to create the statistics page 1580. This page is then returned to the user 1590.

Detailed Description Text (191):
This system provides one central storage place for a person's profile. This storage place is a server available through the public Internet, accessible by any device that is connected to the Internet and has appropriate access. Because of the ubiquitous accessibility of the profile, numerous access devices can be used to customize services for the user based on his profile. For example, a merchant's web site can use this profile to provide personalized content to the user. A Personal Digital Assistant (PDA) with Internet access can synchronize the person's calendar, email, contact list, task list and notes on the PDA with the version stored in the Internet site. This enables the person to only have to maintain one version of this data in order to have it available whenever it is needed and in whatever formats it is needed.

Detailed Description Text (192):
FIG. 17 presents the detailed logic associated with the many different methods for accessing this centrally stored profile. The profile database 1710 is the central storage place for the users' profile information. The profile gateway server 1720 receives all requests for profile information, whether from the user himself or merchants trying to provide a service to the user. The profile gateway server is responsible for ensuring that information is only given out when the profile owner specifically grants permission. Any device that can access the public Internet 1730 over TCP/IP (a standard network communications protocol) is able to request information from the profile database via intelligent HTTP requests. Consumers will be able to gain access to services from devices such as their televisions 1740, mobile phones, Smart Cards, gas meters, water meters, kitchen appliances, security systems, desktop computers, laptops, pocket organizers, PDAs, and their vehicles, among others. Likewise, merchants 1750 will be able to access those profiles (given permission from the consumer who owns each profile), and will be able to offer customized, personalized services to consumers because of this.

Detailed Description Text (193):
One possible use of the ubiquitous profile is for a hotel chain. A consumer can carry a Smart Card that holds a digital certificate uniquely identifying him. This Smart Card's digital certificate has been issued by the system and it recorded his profile information into the profile database. The consumer brings this card into a hotel chain and checks in. The hotel employee swipes the Smart Card and the consumer enters his Pin number, unlocking the digital certificate. The certificate is sent to the profile gateway server (using a secure transmission protocol) and is authenticated. The hotel is then given access to a certain part of the consumer's profile that he has previously specified. The hotel can then retrieve all of the consumer's billing information as well as preferences for hotel room, etc. The hotel can also access the consumer's movie and dining preferences and offer customized menus for both of them. The hotel can offer to send an email to the consumer's spouse letting him/her know the person checked into the hotel and is safe. All transaction information can be uploaded to the consumer's profile after

the hotel checks him in. This will allow partners of the hotel to utilize the information about the consumer that the hotel has gathered (again, given the consumer's permission).

Detailed Description Text (196):
FIG. 18 discloses the detailed interaction between a consumer and the integrator involving one supplier. The user accesses a Web Browser 1810 and requests product and pricing information from the integrator. The request is sent from the user's browser to the integrator's Web/Application Server 1820. The user's preferences and personal information is obtained from an integrator's customer profile database 1830 and returned to the Web/Application server. The requested product information is extracted from the supplier's product database 1840 and customized for the particular customer. The Web/Application server updates the supplier's customer information database 1850 with the inquiry information about the customer. The product and pricing information is then formatted into a Web Page 1860 and returned to the customer's Web Browser.

Detailed Description Text (198):
A suite of software agents running on the application and web servers are programmed to take care of repetitive or mundane tasks for the user. The agents work according to rules set up by the user and are only allowed to perform tasks explicitly defined by the user. The agents can take care of paying bills for the user, filtering content and emails, and providing a summary view of tasks and agent activity. The user interface for the agent can be modified to suit the particular user.

Detailed Description Text (199):
FIG. 19 discloses the logic in accordance with a preferred embodiment processing by an agent to generate a verbal summary for the user. When the user requests the summary page 1900, the server gets the user's agent preferences 1920, such as agent type, rules and summary level from the user profile database 1930. The server gets the content 1940, such as emails, to do list items, news, and bills, from the content database 1950. The agent parses all of this content, using the rules stored in the profile database, and summarizes the content 1960. The content is formatted into a web page 1970 according to a template. The text for the agent's speech is generated 1980, using the content from the content database 1990 and speech templates stored in the database. This speech text is inserted into the web page 1995 and the page is returned to the user 1997.

Detailed Description Text (218):
FIG. 24 is a block diagram of an active knowledge management system in accordance with a preferred embodiment. The system consists of the following parts: back-end 2400 connection to one or more servers, personal mobile wireless clients (Awareness Machine)2430, 2436, public clients (Magic Wall) 2410, 2420, web clients 2446, 2448, e-mail clients 2450, 2460.

Detailed Description Text (219):
Back-end Server (2400) Processes

Detailed Description Text (220):
FIG. 25 is a block diagram of a back end server in accordance with a preferred embodiment. The back-end (2400 of FIG. 24) is a computer system that has the following software active: Intelligent Agents Coordinator (Munin) 2580, Information Prioritization Subsystem 2530, a set of continuously and periodically running information gathering and processing Intelligent Agents 2500, 2502 and 2504, User Profiles Database 2542 and supporting software, Information Channels Database 2542 and supporting software, communications software 2550, information transformation software 2560, and auxiliary software.

Detailed Description Text (226):

Computer system 2640 connected to the back-end server

Detailed Description Text (247):
Other Clients

Detailed Description Text (248):
The Web client is a standard browser navigating to a set of Web pages which allow user to see the same information that is available via the Magic Wall.

Detailed Description Text (249):
The e-mail client is any standard e-mail program.

Detailed Description Text (263):
Most people are mobile throughout their day. The Intelligent Agent Coordinator tries to be sensitive to this fact by attempting to determine, both by observation (unsupervised learning) and from cues from the environment, where users are or are likely to be located. This is certainly important for determining where to send the user's information, but also for determining in which format to send the information. For instance, if a user were at her desk and using the web client, the Intelligent Agent Coordinator would be receiving indications of activity from her PC and would know to send any necessary information there. In addition, because desktop PCs are generally quite powerful, a full-featured, graphically intense version could be sent. However, consider an alternative situation: the Intelligent Agent Coordinator has received an indication (via the keycard reader next to the exit) that you have just left the building. Minutes later the Intelligent Agent Coordinator also receives notification that you have received an urgent message. The Intelligent Agent Coordinator, knowing that you have left the building and having not received any other indications, assumes that you are reachable via your handheld device (for which it also knows the capabilities) and sends the text of the urgent message there, rather than a more graphically-oriented version.

Detailed Description Text (273):
The following code is written and executed in the Microsoft Active Server Pages environment in accordance with a preferred embodiment. It consists primarily of Microsoft Jscript with some database calls embedded in the code to query and store information in the database.

Detailed Description Paragraph Table (5):
PAT PAT GRP # PATTERN EXAMPLE
1 a $PEOPLE$ of Paul Maritz of Microsoft $COMPANY$ b $PEOPLE$ from Bill Gates, Paul Allen and $COMPANY$ Paul Maritz from Microsoft 2 a $TOPIC.sub.-- UPPER$ meeting Push Technology Meeting b $TOPIC.sub.-- UPPER$ mtg Push Technology Mtg c $TOPIC.sub.-- UPPER$ demo Push Technology demo d $TOPIC.sub.-- UPPER$ Push Technology interview interview e $TOPIC.sub.-- UPPER$ Push Technology presentation presentation f $TOPIC.sub.-- UPPER$ visit Push Technology visit g $TOPIC.sub.-- UPPER$ briefing Push Technology briefing h $TOPIC.sub.-- UPPER$ Push Technology discussion discussion i $TOPIC.sub.-- UPPER$ Push Technology workshop workshop j $TOPIC.sub.-- UPPER$ prep Push Technology prop k $TOPIC.sub.-- UPPER$ review Push Technology review l $TOPIC.sub.-- UPPER$ lunch Push Technology lunch m $TOPIC.sub.-- UPPER$ project Push Technology project n $TOPIC.sub.-- UPPER$ projects Push Technology projects 3 a $COMPANY$ corporation Intel Corporation b $COMPANY$ corp. IBM Corp c $COMPANY$ systems Cisco Systems d $COMPANY$ limited IBM limited e $COMPANY$ ltd IBM ltd 4 a about $TOPIC.sub.-- ALL$ About intelligent agents technology b discuss $TOPIC.sub.-- ALL$ Discuss intelligent agents technology c show $TOPIC.sub.-- ALL$ Show the client our intelligent agents technology d re: $TOPIC.sub.-- ALL$ re: intelligent agents technology e review $TOPIC.sub.-- ALL$ Review intelligent agents technology f agenda The agenda is as follows: cleanup clean up cleanup g agenda: $TOPIC.sub.-- ALL$ Agenda: demo client intelligent agents technology demo ecommerce 6 a w/ $PEOPLE$ of Meet w/Joe Carter of $COMPANY$ Andersen Consulting b w/$PEOPLE$ from

Meet w/Joe Carter from $COMPANY$ Andersen Consulting 6 a w/$COMPANY$ per Talk w/Intel per Jason

Detailed Description Paragraph Table (13):

```
_____ ___ ->--
#include file="include/check.sub.-- authentication.inc" <HTML> <HEAD>
<TITLE>mySite! Intentions List</TITLE> <SCRIPT LANGUAGE="JavaScript"> function
intentionsList ( ) { this internalArray = new Array ( ); <% // establish connection
to the database objConnection = Server.CreateObject("ADODB.Connection");
objConnection.Open("Maelstrom"); // create query intentionsQuery =
objConnection.Execute("SELECT * FROM intentions ORDER BY intention.sub.-- name
asc"); %> // write out the options <% numOptions = 0 while (!intentionsQuery.EOF)
{ intentionName = intentionsQuery("intention.sub.-- name"); intentionIcon =
intentionsQuery("intention.sub.-- icon"); %> this.internalArray[<%= numOptions%>] =
new Array(2); this.internalArray[<%= numOptions%>] [0] = "<%= intentionName %>";
this.internalArray[<%= numOptions%>] [1] = "images/<%= intentionIcon %>"; <%
numOptions++; intentionsQuery.moveNext( ); %> <% } %> numIntentions = <%=
numOptions%>; intentionArray = new intentionsList( ).internalArray; function
selectIntention ( ) { for (i=0;i<numIntentions;i++) { if
(IntentionsListSelect.options[i].selected) { intentionNameTextField.value =
intentionArray[i] [0]; //intentionPicture.src = intentionArray [i] [1];
break; } } } </SCRIPT> </HEAD> <BODY BGCOLOR="<%=Session("main.sub.-- background")%
>" style="font-family: Arial"> <CENTER> -->-- <FORM NAME="intention.sub.-- list">
<TABLE FRAME="BOX" border=0 CELLPADDING="2" CELLSPACING="2"> <TR> <TD COLSPAN="3"
STYLE="font: 20pt arial" ALIGN="CENTER"> <B>Add a mySite! Intention</B> </TD> </TR>
<TR> <TD COLSPAN="3"> </TD> </TR> <TR> <TD width="100"> <font size="-1">Please
Select An Intention You Would Like to Add to Your List</font> </TD> <TD colspan=2>
<SELECT ID="(IntentionsListSelect" NAME="IntentionsListSelect" SIZE="10"
style="font: 9pt Arial;" onClick="selectIntention( )."> <%
intentionsQuery.moveFirst( ); for(j=0;j<numOptions;j++) { %> <OPTION VALUE="<%=
intentionsQuery("intention.sub.-- id") %>" <% if (j == 0) { %> SELECTED <% } %>> <%
= intentionsQuery("intention.sub.-- name") %> <% intentionsQuery.moveNext( ) }
intentionsQuery.moveFirst( ); %> </SELECT> </TD> </TR> <TR> <TD COLSPAN="3"> </TD>
</TR> <TR> <TD width="100"> <font size="-1">Customize the Intention
name</font></TD> <TD COLSPAN=2"> <INPUT TYPE="text" NAME="intentionNameTextField"
ID="intentionNameTextField" SIZE="30" VALUE="<%= intentionsQuery("intentio n.sub.--
name") %>"> </TD> </TR> <TR> <TD COLSPAN="3"> </TD> </TR> <TR> <TD COLSPAN="3"
ALIGN="CENTER"> <INPUT TYPE="button" NAME="intentionOKButton" VALUE=" OK "
SIZE="10" ID="intentionOKButton"
onClick="javaScript:top.opener.top.navframe.addAnIn tention( );">            &n
bsp;&nbs p; <INPUT TYPE="button" NAME="intentionCancelButton" VALUE="Cancel"
SIZE="10" ID="intentionCancelButton" onClick="self.close( );"> </TD> </TR> </TABLE>
-->-- </FORM> </CENTER> <% objConnection.Close( ); %> </BODY> </HTML>
```

Current US Cross Reference Classification (1):
705/26

Other Reference Publication (11):
Dynamic Mobile Data Announces Mobile Server Wireless Solution For Enterprise and internet Access; Mar. 1999, pp. 1-2, Anonymous.

Previous Doc Next Doc Go to Doc#

DOCUMENT-IDENTIFIER: US 6279112 B1
TITLE: Controlled transfer of information in computer networks

Application Filing Date (1):
19961029

Detailed Description Text (16):
Channel objects may be embedded not only in documents or pages on the World Wide Web, but in an alternative implementation they may be embedded in e-mail messages, OLE objects, ActiveX applets, etc. In fact, all of the communications between the server computer and the client computer and between the information source computer and the client computer may occur by e-mail, via compound documents, etc.

Detailed Description Text (18):
Referring to FIGS. 4A and 4B, in operation of the network-based system of FIG. 3, the coupon-providing server sends a document to the client computer containing an embedded digital coupon (step 112). The coupon may be an executable program or program fragment expressed in machine-executable form, such as an ActiveX applet, and protected against unauthorized tampering by means of an authenticator such as a digital signature or MAC code (Message Authentication Code), or the coupon may be a digitally signed set of inputs to a program already residing at the client computer. The coupon contains a set of restrictions such as an expiration date, a product code or item number, and a discount amount. Alternatively, the coupon may simply contain a coded number that can be understood by the smart digital offer object described below.

Detailed Description Text (20):
In one embodiment the coupon registry at the client computer is a purchasing history and the coupons are digital receipts identifying products purchased, dates of purchase, and possibly prices paid, together with authenticators of the digital receipts. The digital receipts function in the same manner as ordinary coupons because they will be used for the purpose of offering an adjusted price (typically a discounted price) to the user of the client computer. These digital receipts are transmitted from a server to the client computer together with authenticators upon completion of a purchase transaction.

Detailed Description Text (21):
The client computer fetches a document of web-based information from the offer-providing server that contains a smart digital offer object (step 118). The smart digital offer object may be an executable program or program fragment expressed in machine-executable form, such as an ActiveX applet, and protected against unauthorized tampering by means of an authenticator such as a digital signature or MAC code, or the smart digital offer object may be a digitally signed set of inputs to a program already residing at the client computer. The smart digital offer object received by the client computer may be protected against unauthorized tampering by means of a digital signature or MAC code. In an alternative embodiment the smart digital offer object remains at the offer providing server and need not be protected against tampering. The client computer activates the smart digital

offer object (step 120), and the smart digital offer object attempts to observe the parameters of the execution environment at the client machine, including the presence of coupons, and possibly other information such as a purchasing history recorded on the client computer.

Detailed Description Text (23):
The terms or conditions of the offer, such as price and payment terms, are calculated by the smart digital offer object using formulas that depend on the information contained in the digital coupons and the other information examined by the smart digital offer object, including the time of day, or user profile information such as membership codes, user's age, user's income, and other demographic information certified by an independent authority with an authenticator. When the user accepts the offer (step 128) the client computer sends a message to the offer-providing server indicating that the user has accepted the offer, or sends the message to an intermediary server that is trusted by the client computer to maintain the confidentiality of user-specific information and is trusted by the offer-providing server to verify the terms on which the offer was accepted (step 130). The message sent to the offer-providing server or the intermediary server includes the terms upon which the offer was accepted and also includes an authenticator. The offer-providing server or the intermediary server verifies the terms on which the offer was accepted by verifying the authenticator (step 132), and, if an intermediary server is used, the intermediary server reports the acceptance of the offer and the terms on which it was accepted to the offer-providing server. The offer-providing server then fulfills the offer by causing the offered product or service to be provided to the user (step 134).

Detailed Description Text (37):
Referring to FIG. 8, in operation of the network-based system of FIG. 7 the client computer first obtains valuable web-based information (step 310) in the form of a document containing an embedded active link that retrieves additional information and also implements a small program or applet. The active link may be embedded in the document by means of the known technique of ActiveX Controls. The client computer displays the document (step 312). When a user clicks on a representation of the active link (step 314) or, in an alternative embodiment described in detail below, when the active link is called by the browser at the client computer (step 316), the client computer activates the active link (step 318). Activation of the active link at the client computer includes activation of the applet (step 320), which may fetch from the server computer, or elsewhere, a machine-executable program that is used for client-side metering of the end-user's access to valuable web-based information, as is explained below. The client computer may store the machine-executable program after it is first retrieved, so that subsequent activations of the applet do not require communication with another computer to obtain the program. Activation of the applet causes the client computer to record in the metering log the fact that a certain document, or a certain portion of the document, has been displayed (step 322).

Previous Doc Next Doc Go to Doc#

L9: Entry 3 of 13 File: USPT Jun 6, 2000

DOCUMENT-IDENTIFIER: US 6073124 A
TITLE: Method and system for securely incorporating electronic information into an online purchasing application

Abstract Text (1):
A method and system for facilitating digital commerce using a secure digital commerce system is provided. The secure digital commerce system is arranged according to a client/server architecture and includes a modularized DCS client and DCS server. The DCS client and the DCS server are incorporated into an online purchasing system, such as a virtual store, to perform the purchase and online delivery of electronic content. The DCS client includes a set of components which include a secured copy of the merchandise and various components needed to license and purchase the merchandise and to unsecure and process (e.g., execute) the licensed merchandise. The DCS client communicates with the DCS server to download the components onto a customer's computer system and to license and purchase a requested item of merchandise. The DCS server, which includes a content supplier server, a licensing and purchasing broker, and a payment processing function, supplies merchandise-specific components and licenses the requested item of merchandise by generating an electronic certificate. The electronic certificate contains license parameters that are specific to the requested merchandise and an indicated purchasing option. Once a valid electronic license certificate for the requested merchandise is received by the DCS client, the merchandise is made available to the customer for use in accordance with the licensing parameters contained in the electronic license certificate.

Application Filing Date (1):
19970715

DATE ISSUED (1):
20000606

Brief Summary Text (9):
To perform digital commerce, today's computer networking environments utilize a client/server architecture and a standard protocol for communicating between various network sites. One such network, the World Wide WEB network, which comprises a subset of Internet sites, supports a standard protocol for requesting and for receiving documents known as WEB pages. This protocol is known as the Hypertext Transfer Protocol, or "HTTP." HTTP defines a high-level message passing protocol for sending and receiving packets of information between diverse applications. Details of HTTP can be found in various documents including T. Berners-Lee et al., Hypertext Transfer Protocol--HTTP 1.0, Request for Comments (RFC) 1945, MIT/LCS, May, 1996, which is incorporated herein by reference. Each HTTP message follows a specific layout, which includes among other information a header, which contains information specific to the request or response. Further, each HTTP message that is a request (an HTTP request message) contains a universal resource identifier (a "URI"), which specifics a target network resource for the request. A URI is either a Uniform Resource Locator ("URL") or Uniform Resource Name ("URN"), or any other formatted string that identifies a network resource. The URI contained in a request message, in effect, identifies the destination machine for a message. URIs, as an example of URls, are discussed in detail in T. Berners-

Lee, et al., Uniform Resource Locators (URL), RFC 1738, CERN, Xerox PARC, Univ. of Minn., December, 1994, which is incorporated herein by reference.

Brief Summary Text (10):
FIG. 1 illustrates how a browser application, using the client/server model of the World Wide WEB network, enables users to navigate among network nodes by requesting and receiving WEB pages. For the purposes of this specification, a WEB page is any type of document that abides by the HTML format. That is, the document includes an "<HTML>" statement. Thus, a WEB page can also be referred to as an HTML document or an HTML page. HTML is a document mark-up language, defined by the Hypertext Markup Language ("HTML") specification. HTML defines tags for specifying how to interpret the text and images stored in an HTML page. For example, there are HTML tags for defining paragraph formats and text attributes such as boldface and underlining. In addition, the HTML format defines tags for adding images to documents and for formatting and aligning text with respect to images. HTML tags appear between angle brackets, for example, <HTML>. Further details of HTML arc discussed in T. Berners-Lee and D. Connolly, Hypertext Markup Language-2.0, RFC 1866, MIT/W3C, November, 1995, which is incorporated herein by reference.

Brief Summary Text (11):
In FIG. 1, a WEB browser application 101 is shown executing on a client computer system 102, which communicates with a server computer system 103 by sending and receiving HTTP packets (messages). The WEB browser application 101 requests WEB pages from other locations on the network to browse (display) what is available at these locations. This process is known as "navigating" to sitès on the WEB network. In particular, when the WEB browser application 101 "navigates" to a new location, it requests a new page from the new location (e.g., server computer system 103) by sending an HTTP-request message 104 using any well-known underlying communications wire protocol. HTTP-request message 104 follows the specific layout discussed above, which includes a header 105 and a URI field 106, which specifies the target network location for the request. When the server computer system machine specified by URI 106 (e.g., the server computer system 103) receives the HTTP-request message, it decomposes the message packet and processes the request. When appropriate, the server computer system constructs a return message packet to send to the source location that originated the message (e.g., the client computer system 102) in the form of an HTTP-response message 107. In addition to the standard features of an HTTP message, such as the header 108, the HTTP-response message 107 contains the requested WEB page 109. When the HTTP-response message 107 reaches the client computer system 102, the WEB browser application 101 extracts the WEB page 109 from the message, and parses and interprets the HTML code in the page (executes the WEB page) in order to display the document on a display screen of the client computer system 102 in accordance with the HTML tags.

Brief Summary Text (15):
Exemplary embodiments of the secure digital commerce system ("DCS") include a DCS client and a DCS server. The DCS client includes a plurality of client components, which are downloaded by a boot program onto a customer computer system in response to requesting an item of merchandise to be licensed or purchased. The downloaded client components include a secured (e.g., encrypted) content file that corresponds to the content of the requested item and licensing code that is automatically executed to ensure that the item of merchandise is properly licensed before a customer is permitted to operate it. The DCS server includes a content supplier server, which provides the DCS client components that are specific to the requested item, and a licensing and purchasing broker, which generates and returns a secure electronic licensing certificate in response to a request to license the requested item of merchandise. The generated electronic license certificate contains licensing parameters that dictate whether the merchandise is permitted to be executed. Thus, once properly licensed, the downloaded client components in conjunction with the electronic license certificate permit a legitimate customer to execute (process) purchased content in a manner that helps prevent illegitimate

piracy.

Brief Summary Text (16):
In one embodiment, the electronic license certificate is generated from tables stored in a password generation data repository. Each table contains fields that are used to generate the license parameters. Each electronic license certificate is generated specifically for a particular item of merchandise and for a specific customer request. Also, the electronic license certificate is secured, such as by encryption, to prevent a user from accessing the corresponding item of merchandise without proper authorization. One technique for securing the electronic license certificate uses a symmetric cryptographic algorithm.

Brief Summary Text (17):
The secure digital commerce system also supports the ability to generate emergency electronic license certificates in cases where an electronic license certificate would not normally be authorized. To accomplish this objective, a separate emergency password generation table is provided by the password generation data repository. In addition, the secure digital commerce system reliably downloads the client components even when a failure is encountered during the download procedure. Further, a minimum number of components are downloaded.

Brief Summary Text (18):
In addition to generating electronic license certificates, the licensing and purchasing broker may also include access to a payment processing function, which is invoked to authorize a particular method of payment for a particular transaction. The licensing and purchasing broker may also include access to a clearinghouse function used to track and audit purchases.

Brief Summary Text (19):
Digital commerce is performed using the secure digital commerce system as follows. A customer invokes an online purchasing system to request an item of merchandise and to indicate a purchasing option (such as "try" or "buy"). The DCS client then downloads onto a customer computer system the client components that are associated with the requested item. Included in these components is a secured content component. The secured content component is then installed and executed (processed) in a manner that automatically invokes licensing code. The licensing code, when the requested item is not yet licensed properly, causes the requested item to be licensed by the licensing and purchasing broker in accordance with the indicated purchasing option before the content component becomes operable. Specifically, the licensing and purchasing broker generates a secure electronic license certificate and completes an actual purchase when appropriate. The broker then returns the electronic license certificate to the licensing code, which unsecures (e.g., unencrypts) and deconstructs the electronic license certificate to determine the licensing parameters. The licensing code then executes (processes) the content component in accordance with the license parameters.

Drawing Description Text (2):
FIG. 1 illustrates how a browser application, using the client/server model of the World Wide WEB network, enables users to navigate among network nodes by requesting and receiving WEB pages.

Drawing Description Text (6):
FIG. 5 is a block diagram of a general purpose computer system for practicing embodiments of the DCS client.

Drawing Description Text (7):
FIG. 6 is an example flow diagram of the steps performed to generate the components of the DCS client.

Drawing Description Text (9):

FIG. 8 is an example flow diagram of the steps performed by a boot program executed on a customer computer system to download client components when licensing a selected item of merchandise.

Drawing Description Text (19):
FIG. 18 is an example display screen for indicating that a purchasing transaction has been authorized.

Drawing Description Text (20):
FIG. 19 is an example block diagram that illustrates one technique for ensuring secure communications between a DCS client component and a licensing and purchasing broker.

Drawing Description Text (21):
FIG. 20 is an example encrypted message protocol for sending encrypted messages between a DCS client component and a licensing and purchasing broker.

Detailed Description Text (4):
Each icon is typically linked to a server site on the network, which is responsible for supplying the content of the item when purchased if the item is capable of electronic delivery. When the user selects one of the icons, the browser application, as a result of processing the link, sends a request for the selected item to the server site. Thus, when a customer selects the icon 203, an HTTP request message is sent to an appropriate server site to locate and download the software modules that correspond to "RETURN OF ARCADE."

Detailed Description Text (6):
The secure digital commerce system is arranged according to a client/server architecture and provides a modularized DCS client and a modularized DCS server that interact with the online purchasing system to perform a purchase. The DCS client includes a set of client components; support for downloading the client components onto a customer computer system; and support for communicating with the DCS server to license an item of merchandise. The client components contain a secured (e.g., encrypted) copy of the content and various components needed to license and purchase the merchandise and to unsecure (e.g., decrypt) and execute the licensed merchandise. The DCS client communicates with the DCS server to download the client components onto a customer's computer system in response to a request for merchandise from the online purchasing system. The DCS client also communicates with the DCS server to license and purchase the requested merchandise. The DCS server generates an electronic license certificate, which contains license parameters (e.g., terms) that are specific to the requested merchandise and to a desired purchasing option (such as trial use, permanent purchase, or rental). The DCS server then sends the generated electronic license certificate to the DCS client. Once a valid electronic license certificate for the requested merchandise is received by the DCS client, the merchandise is made available to the customer for use in accordance with the license parameters contained in the electronic license certificate.

Detailed Description Text (7):
The DCS client includes a download file, a user interface library, a purchasing library, a secured content file, a DCS security information file, and licensing code. There is a download file for each item of merchandise that can be distributed electronically, which contains an executable boot program. The boot program is responsible for determining what components need to be downloaded for a requested item of merchandise. The secured content file contains the content that corresponds to the requested item of merchandise. The content may be a computer program, data, or a combination of both. For the purposes of this specification, "secure" or "secured" implies the use of cryptography or other types of security, including the use of hardware. One or more of the remaining components can be shared by several items of merchandise. For example, the user interface library, which defines a user

interface used to purchase and license merchandise, may be specific to an item of merchandise or may be uniform for an entire online purchasing system. The purchasing library, licensing code, and DCS security information file are used to interact with the DCS server to properly license requested merchandise. In particular, the licensing code ensures that the requested merchandise is not operable by the customer until it has been properly licensed by the DCS server.

Detailed Description Text (8):
The DCS server includes a content supplier server, a licensing and purchasing broker, and a payment processing function. The content supplier server provides the merchandise-specific DCS client components. The licensing and purchasing broker generates electronic license certificates and manages purchases. The payment processing function authorizes payment for a particular transaction. One or more of each of these entities may be available in a DCS server.

Detailed Description Text (10):
For the purposes of this specification, any client/server communication architecture and communication protocol that supports communication between the DCS client and the DCS server could be used.

Detailed Description Text (12):
FIG. 3 is an overview block diagram of the secure digital commerce system. FIG. 3 includes a DCS client 301 and a DCS server 302, which are used with an online purchasing application, such as a WEB browser application 303, to provide a purchasing interface for a potential customer. The DCS client 301 includes a virtual store 304 and a data repository 305. The virtual store 304 provides a customer front end 312 and stores in the data repository 305 merchandise-specific download files 313. The customer front end 312 includes WEB pages and associated processing support, which are downloaded onto a customer computer system 311 to enable a user to purchase merchandise. The download files 313, which each contain an executable boot program and a component list, are used to download the merchandise-specific client components (for example, a secured content file and licensing code). When an item of merchandise is requested, the associated download file is processed to extract the executable boot program and the component list. The executable boot program downloads the needed components from the content supplier server 306 using the component list, which specifies the components that are needed to successfully license and operate the corresponding item of merchandise. In an alternate embodiment, download files are generated dynamically from component lists, which lists are stored in the data repository 305.

Detailed Description Text (13):
The DCS server 302 includes a content supplier server 306, a licensing and purchasing broker (server) 307, a password generation data repository 308, and a payment processing function 309. The licensing and purchasing broker 307 includes a separate licensing library 310 (passgen.dll), which contains the code for generating an appropriate license in response to a request from the virtual store. The licensing library 310 uses the password generation data repository 308 to generate an electronic license

Detailed Description Text (14):
certificate ("ELC") with licensing parameters that correspond to a particular item of merchandise. An electronic license certificate is encrypted electronic data that provides information that can be utilized to determine whether a particular customer is authorized to execute the merchandise. Such information may include, for example, the specification of a period of time that a particular customer is allowed to execute the merchandise for trial use. The data repository 308 contains tables and fields that are used to create the license parameters of a license. The data repository 308 may contain information that is supplied by the source companies of the available merchandise. The payment processing functions 309 are used by the licensing and purchasing broker 307 to charge the customer and tc

properly credit the appropriate supplier when the customer requests an actual purchase (rather than trial use or another form of licensing). In addition, clearinghouse functions may be invoked by the licensing and purchasing broker 307 to audit and track an online purchase. Clearinghouse functions may be as provided by well-known commercial sources, such as Litlenet and Cybersource. Similarly, payment processing functions may be provided using well-known commercial credit card authorization services.

Detailed Description Text (15):
FIG. 4 is an overview flowchart of the example steps performed by the secure digital commerce system components to perform the licensing and purchase of electronic data. This figure briefly describes the interactions between the components shown in FIG. 3 to accomplish the downloading, licensing, and purchasing of a requested item of merchandise when it can be delivered online. In step 401, the potential customer downloads a WEB page (part of the customer front end 312) from the virtual store 304 that includes the item to be requested (see, for example, FIG. 2). In step 402, the customer requests an item of merchandise, for example, by selecting an icon that is linked to a download file that corresponds to the desired item. In response to the selection, in step 403, the virtual store 304 downloads and installs the download file, which extracts the executable boot program and component list and causes execution (preferably as a background task) of the executable boot program on the customer computer system 311. In step 404, the boot program reads the component list to determine what DCS client components to download and requests the determined components from the appropriate contents supplier server 306. The component list, as further described below with reference to Table 2, indicates source and target locations for each component to be downloaded. In step 405, the boot program installs a downloaded (secured) content file that is associated with the desired item of merchandise and causes the content file to be processed (executed). When the content file is a computer program, then the downloaded content file has been previously configured to automatically cause licensing code to be executed before the content file is executed. When instead the content file is data to be input to a computer program, then the content player is previously configured to automatically cause the licensing code to be executed first before the content file data is processed. More specifically, the downloaded content player is installed by the boot program to process the secured (e.g., encrypted) content file data. The boot program then starts the execution of the content player, which invokes and causes execution of the downloaded licensing code. Thus, in step 406, the licensing code, which is incorporated into either the content file or the content player, is executed. In step 407, if the licensing code determines that a valid ELC already exists, then the content file continues to be processed in step 412, else the licensing code continues in step 408. In step 408, the licensing code requests a valid ELC from the licensing and purchasing broker 307. In step 409, the licensing and purchasing broker 307 determines whether a purchase is requested and, if so, continues in step 410, else continues in step 411. In step 410, the licensing and purchasing broker 307 obtains a method for payment and authorizes the payment method using the payment processing function 309. In step 411, the licensing and purchasing broker 307 generates an appropriate ELC using the licensing library 310 and the password generation data repository 308 and returns the generated EL-C to the licensing code. In step 412, if portions of the content file are encrypted as will be further described, then the content file is decrypted and processed.

Detailed Description Text (16):
As indicated above, when the downloaded (secured) content file is a computer program, licensing code is automatically invoked to verify the existence of, or obtain, a valid electronic license certificate for a requested item and to decrypt and execute the content file. One mechanism for incorporating licensing code into a content file such that it is automatically invoked is discussed in detail with reference to related U.S. patent application Ser. No. 08/792,719, entitled "Method and System for Injecting New Code Into Existing Application Code," filed on Jan.

29, 1997. That patent application describes a technique for inserting licensing code into an existing application and for inserted security code that securely executes the application code. The security code uses an incremental decryption process to ensure that a complete version of the unmodified application code is never visible at any one time (to avoid illegitimate copying). Thus the security code mechanism described therein makes it impossible for someone to create an unmodified version of the application in a reasonable amount of time. The insertion technique described therein can be used to insert into a content file the licensing code component of the DCS client, which communicates with the licensing and purchasing broker to generate an ELC. Further, the encryption/decryption technique described therein may be used in the current context to incorporate security code that securely decrypts and executes the downloaded content file.

Detailed Description Text (18):
In exemplary embodiments, the DCS client is implemented on a computer system comprising a central processing unit, a display, a memory, and other input/output devices. Exemplary embodiments of the DCS client are designed to operate in a globally networked environment, such as a computer system that is connected to the Internet. FIG. 5 is a block diagram of a general purpose computer system for practicing embodiments of the DCS client. The computer system 501 contains a central processing unit (CPU) 502, a display 503, a computer memory (memory) 505, or other computer-readable memory medium, and other input/output devices 504. Downloaded components of the DCS client preferably reside in the memory 505 and execute on the CPU 502. The components of the DCS client are shown after they have been downloaded and installed on the computer system 501 by an executable boot program and after an appropriate electronic license certificate has been generated and installed. Specifically, the components of the DCS client include the executable boot program 507 (SAFEboot); a user interface library 508 (SAFEUI.dll); a purchasing request library 509 (SAFEBuy.dll); an encrypted content file 510, which is shown with incorporated licensing code 511 (SAFE.dll); an encrypted DCS security information file 512, which is associated with the encrypted content file 510; and an electronic licensing certificate 514 (ELC). As shown, each library is typically implemented as a dynamic link library (a "DLL"). In addition to these components, when the encrypted content file contains data that is not a computer program, the memory 505 contains a content player 513 for processing the content file 510, which has incorporated licensing code 511. Also, WEB browser application code 506 is shown residing in the memory 505. Other programs 515 also reside in the memory 505. One skilled in the art will recognize that exemplary DCS client components can also be implemented in a distributed environment where the various programs shown as currently residing in the memory 505 are instead distributed among several computer systems. For example, the encrypted content file 510 may reside on a different computer system than the boot program 507.

Detailed Description Text (19):
In exemplary embodiments, the DCS server is implemented on one or more computer systems, each comprising a central processing unit, a memory and other input/output devices. Each of these computer systems may be a general purpose computer system, similar to that described in FIG. 5, which is connected to a network. The server systems that comprise the server portion may or may not include displays. The password generation data repository may be implemented using any well-known technique for implementing a database or any other type of data repository. Although shown as a separate facility, one skilled in the art will recognize that the data repository may be incorporated as a component of the computer system that is used to implement the licensing and purchasing broker. Further, one skilled in the art will also recognize that a variety of architectures are possible and can be used to implement exemplary embodiments of the DCS server.

Detailed Description Text (20):
FIG. 6 is an example flow diagram of the steps performed to generate the components of the DCS client. In an exemplary embodiment, these steps are performed by a

utility program referred to as the SAFEmaker utility. The SAFEmaker utility is responsible for generating the downloadable components that correspond to an item to be supplied as online merchandise. In addition, the utility generates a secured content file that can only be processed when access is granted. This capability is referred to as making the file "SAFE" (hence, the SAFE-prefix in the component names). Making a content file "SAFE" implies that security code and licensing code are incorporated into the content file (or content player, in the case of digital content that is not a computer program) to ensure that the online merchandise is usable only when proper licensing has been performed. Typically, this process involves encrypting some portion of the content file. Some components generated by the SAFEmaker utility are stored on the content supplier server (e.g., content supplier server 306 in FIG. 3) and are downloaded in response to requests from the virtual store front end. Other components are stored on the virtual store, which may be located on a different computer system from the content supplier server. The SAFEmaker utility also updates the password generation data repository of the DCS server with merchandise-specific information.

Detailed Description Text (21):
Specifically, in step 601, the utility incorporates licensing and security code into the supplier specific electronic content or content player. As described above, an exemplary embodiment incorporates licensing and security code according to the techniques described in the related U.S. patent application Ser. No. 08/792,719, entitled "Method and System for Injecting New Code into Existing Application Code," filed on Jan. 29, 1997 or by calling routines of an API as appropriate (e.g., when a content player is needed). One skilled in the art, however, will recognize that any technique for ensuring that proper licensing code gets executed when the content is processed and for encrypting (and subsequently decrypting) the content file will operate with embodiments of the present invention. In step 602, the utility produces one or more files that contain the (partially or fully) encrypted content. In step 603, the utility produces an encrypted DCS security information file(s), which contain information that is used, for example, to decrypt the content and to produce a proper license. The contents of an encrypted DCS security information file are described in further detail below with reference to Table 1. In step 604, the utility creates a component list file (an ".ssc" file) and a download file for this particular online merchandise. Specifically, in an embodiment that statically generates download files, a self-extracting installation file is generated (the download file), which contains the component list file (an ".ssc" file) specific to the merchandise and the executable boot program. As described above, the download file, which contains the executable boot program and the component list, is typically stored on the virtual store computer system. The executable boot program uses the component list file to determine the components to download and to download them when particular electronic content is requested. An example component list file is described further below with reference to Table 2. In step 605, the utility stores the download file on the virtual store computer system (e.g., virtual store 304 in FIG. 3). When instead the download files are dynamically generated by the virtual store when needed for a particular WEB page, then in steps 604 and 605, the utility creates and stores only the component list file. In step 606, the utility stores the other components of the DCS client, for example, the encrypted content and DCS security information files, the licensing code, and the user interface library on the content supplier server system (e.g., content supplier server 306 in FIG. 3). In step 607, the utility updates the password generation data repository (e.g., password generation database 308 in FIG. 3) with the merchandise-specific licensing information, for example, the fields used to generate the license parameters of a valid electronic license certificate, and then returns. An example password generation data repository is discussed in further detail with reference to Tables 3, 4, and 5. One skilled in the art will recognize that the generation of these components and the password generation data may be performed at different times and by separate utilities.

Detailed Description Text (23):
data in the encrypted DCS security information file is encrypted separately from the content file to enable multiple items of merchandise to share purchasing, licensing, and decryption information. This capability is especially useful when the items are provided by the same content supplier server. Thus, a single encrypted DCS security information file may be associated with more than one encrypted content file. In addition, each field in the DCS security information file is encrypted separately. By separately encrypting each field, purchasing or licensing information can be changed without having to re-encrypt the content file or the rest of the DCS security information file.

Detailed Description Text (26):
Each entry contains a tag that specifies how to process the component when it is downloaded and sufficient information to download a component if the file indicated by the TRIGGER field is not already present on the customer computer system. Specifically, the tag (in this example "Execute") specifies what to do with the component referred to by the LOCAL field once it is downloaded. An "Execute" tag specifies that the component referred to by the LOCAL field (e.g., "setup.exe") will always be executed. A "Component" tag specifies that the component referred to by the LOCAL field is to be downloaded with no further processing. An "ExecuteOnce" tag specifies that the component referred to by the LOCAL field is to be executed only if the file referred to by the TRIGGER field does not already exist. The TRIGGER field of each entry indicates the location of a file that is present when the component does not need to be downloaded. Thus, the TRIGGER field is used to determine whether to download a component. The URI field indicates the location of a content supplier server that can provide the component. In addition, the MSGDIG field contains a message digest, which is used to determine whether the component has been successfully loaded. Use of the message digest is described in further detail below with respect to FIG. 8. The ProductUUID, NAME, and DESCRIPTION fields indicate identifying information used by the licensing code. When present, these fields are typically stored in a system registry and used by the licensing code to determine which DCS security information file to use for a particular content file. In addition, the NAME field may be displayed by the boot program executable to give user feedback regarding the component currently being downloaded. The LOCAL field indicates a target location for the downloaded component on the customer computer system.

Detailed Description Text (29):
FIG. 8 is an example flow diagram of the steps performed by a boot program executed on a customer computer system to download client components when licensing a selected item of merchandise. (These steps correspond to steps 404-405 in FIG. 4.) The boot program is implemented such that it downloads only the components that are necessary to license (and optionally purchase) the selected item. For example, if the user interface library to be used to purchase the selected item is the same library as one already downloaded, then it is not downloaded again. In addition, the boot program can recover from a failure during the load process and can resume downloading where it left off. The boot program accomplishes these objectives by using a message digest algorithm to determine whether a component has been successfully downloaded onto a customer computer system.

Detailed Description Text (30):
Specifically, in step 801, the boot program reads the component list (the ".ssc" file) associated with the selected item of merchandise to determine what components to download from a specified content supplier server. In steps 802-808, the boot program executes a loop to process each remaining component in the component list that has not already been successfully downloaded. Specifically, in step 802, the boot program selects the next component from the component list that appears following the last successfully read component. In step 803, the boot program determines whether all of the remaining components of the list have been processed, and if so, returns, else continues in step 804. In step 804, the boot program

determines whether the file indicated by the TRIGGER field is already present. If not, the boot program obtains the component indicated by the URI value from the content supplier server and stores the obtained component as indicated by the LOCAL value (see Table 2). In step 805, the boot program calculates a message digest (the value of a one-way hash function) for the downloaded component. In step 806, the determined message digest for the newly downloaded component is compared with a previously stored message digest in the component list (see the MSGDIG value in Table 2). In an exemplary embodiment, an MD5 algorithm is used to calculate a message digest. However, one skilled in the art will recognize that any message digest algorithm or any function capable of determining a predictable value for the downloaded component for comparison to an already stored value may be used. The MD4 and MD5 algorithms are described in Bruce Schneier, Applied Cryptography, John Wiley & Sons, Inc., 1994, which is hereby incorporated by reference. In step 807, if the calculated message digest is identical to the stored message digest, then the boot program continues in step 808, else continues back to the beginning of the loop in step 802, because a failure has occurred in downloading the component. In step 808, the boot program sets an indicator of the last successfully read component to indicate the component most recently loaded. In step 809, the boot program processes the component according to the tag (e.g., "Execute"), and continues back to step 802 to select the next component to download. Note that the tag associated with each component entry will automatically cause the secured content file (or the content player, depending on the situation) to begin executing.

Detailed Description Text (37):
FIG. 11 is an example flow diagram of the steps performed by licensing code to determine whether a valid electronic licensing certificate is available. In step 1101, the code retrieves, decrypts, and decodes the electronic licensing certificate (ELC) to obtain the parameters of the license (e.g., the license terms). The license parameters that are obtained in step 1101 indicate, for example, how many uses of a particular license can be executed or, for example, how many different user passwords are able to use the same electronic license. In addition, license parameters that reflect an authorized time period for use may be specified. In step 1102, the code tests various attributes of the customer computer system to determine whether the conditions indicated by the retrieved license parameters have been met. In step 1103, if all of the conditions have been met (for example, the license use period has not expired), then the code returns indicating that a valid license is in effect. Otherwise, the code returns indicating that the current license is invalid.

Detailed Description Text (38):
In an exemplary embodiment, the ELC is encrypted and decrypted using a symmetric key algorithm. A symmetric algorithm implies that the same key is used to encrypt a plaintext message and to decrypt a ciphertext message. Any symmetric key algorithm could be used. Symmetric and public key cryptography, both of which are utilized by exemplary embodiments of the present invention, are described in detail in Bruce Schneier, Applied Cryptography, John Wiley & Sons, Inc., 1994, which is herein incorporated by reference. According to one technique, the DeveloperID and SecretKey fields (stored in the encrypted information file) are used to formulate a symmetric key, which is client and product specific. These fields are provided by the supplier when the SAFEmaker utility is executed to produce the components of the DCS client (see FIG. 6). Because the encryption of the ELC is provided by the licensing and purchasing broker and the corresponding decryption of the ELC is provided by the licensing code, the encryption and decryption code are preferably synchronized to correspond to one another. For this reason, a separate dynamic link library (e.g., passgen.dll) is used by the licensing and purchasing broker to allow the encryption algorithm to be replaced at any time to correspond to different licensing code.

Detailed Description Text (40):

Specifically, in step 1201, the broker determines whether a buy request has been received and, if so, continues in step 1202, else continues in step 1206. In step 1202, the broker causes the licensing code (specifically, the user interface library routines) executing on the customer computer system to obtain credit card or purchase order information if such information was not already sent with the request. A sample user interface for obtaining method of payment information and for verifying the purchase transaction are described below with reference to FIGS. 14-17. Once the credit card or purchase order information has been obtained by the licensing and purchasing broker, then in step 1203 the broker obtains payment authorization from a payment processor such as the payment processing function 309 in FIG. 3 and informs the licensing code accordingly. One skilled in the art will recognize that any mechanism for is authorizing use of a credit card could be used. In step 1204, the customer's credit card account is charged, and the supplier system is automatically credited. One skilled in the art will recognize that the licensing and purchasing broker can either credit the supplier directly at this time by sending the appropriate information to the credit card company, or can have the credit card company pay the licensing and purchasing broker, which in turn is responsible for payment to the supplier. In step 1205, the broker informs the licensing code of payment authorization and continues in step 1207. An example user interface for reporting the transaction identification information to the customer is described below with reference to FIG. 18. If payment has not been authorized, then the broker returns such information to the licensing code, discontinues execution of the steps in FIG. 12, and fails to generate a valid ELC.

Detailed Description Text (44):
FIGS. 14-17 provide sample user interface display screens that are displayed by the licensing code (via the user interface library) to retrieve method of payment information. These display screens may be presented in response to requests from the licensing and purchasing broker for more information. The particular display screens presented are determined by the user interface library that is associated with the downloaded content file or by a default user interface available for the virtual store (see e.g., SAFEUI.dll 508 in FIG. 5). As mentioned, the appropriate user interface library is determined by the licensing code from the UILibName field of the DCS security information file. FIG. 14 is an example display screen for selecting a particular credit card. FIG. 15 is an example display screen for entering a password for a selected credit card. The credit card data is sent to the licensing and purchasing broker in encrypted form. In an exemplary embodiment, the credit card information is stored on the customer computer system using a secure technique. One such technique is known as "wallet technology." Wallet technology is an ActiveX control supplied by Microsoft Corp., which encrypts credit card information on a client's hard disk and keeps track of all credit cards. FIG. 16 is an example display screen for adding a new credit card. FIG. 17 is an example display screen for allowing a customer to verify an intent to purchase after supplying a method of payment. The display screen includes pricing information, which is supplied to the licensing code by the licensing and purchasing broker using the password generation data repository. Once the user has selected the Buy pushbutton 1702 in FIG. 17 indicating agreement to purchase the merchandise at the displayed price, the credit card (or purchase order) information is forwarded to the licensing and purchasing broker. FIG. 18 is an example display screen for indicating that a purchasing transaction has been authorized by the licensing and purchasing broker and the particular transaction identifier.

Detailed Description Text (45):
Communications between the DCS client components and the licensing and purchasing broker are preferably performed using a secure communication methodology. FIG. 19 is an example block diagram that illustrates one technique for ensuring secure communication between a DCS client component and a licensing and purchasing broker. Although FIG. 3 may imply that the downloaded components communicate with the licensing and purchasing broker to request licensing and purchasing and to receive the generated ELC, one skilled in the art will recognize that it is also possible

for these components to communicate via a server associated with the virtual store. In FIG. 19, communication between the client components (clients) 1901 and 1902 and the licensing and purchasing broker 1903 depends upon secure key exchange. Secure key exchange is accomplished by sending a client-specific symmetric key using a public/private key algorithm. The client-specific symmetric key is used solely for communication between that client and the licensing and purchasing broker. Specifically, a separate communication session-specific symmetric key is provided by each client for each communication session and is sent to the licensing and purchasing broker 1903 in a session initiation message using the broker's public key. One technique for distributing and obtaining the broker's public key is to use a commercially available digital signature service, such as Verisign. Because the broker 1903 is the only process that knows its own private key, the broker 1903 decrypts the session initiation message using its private key and retrieves the client's session-specific symmetric key. Thereafter, all messages from the broker 1903 to the client 1901 are encrypted by the broker 1903 using the client 1901's symmetric key. Client 1901 is then able to decrypt a received message using the symmetric key that it initially generated and sent to the broker 1903. Client 1901 encrypts messages to send to the broker 1903 also using client 1901's symmetric key. Similarly, the client 1902 sends its own encrypted symmetric key to broker 1903 using the broker's public key. The broker 1903 in turn communicates with the client 1902 using the client-specific symmetric key that corresponds to client 1902.

Detailed Description Text (47):
FIG. 20 is an example encrypted message data structure for sending encrypted messages between a DCS client component and a licensing and purchasing broker. Plaintext message 2001 is encrypted as specified in FIG. 19 and stored according to the layout of ciphertext message 2002. Ciphertext message 2002 contains a message digest 2003 and an encrypted symmetric key 2004, which has been encrypted using the licensing and purchasing broker's public key. In addition, field 2005 contains the message content, which has been encrypted using the symmetric key that is sent in encrypted form in field 2004.

Detailed Description Text (51):
Table 5 is an example emergency password table. An emergency password table is used by the licensing and purchasing broker to generate an emergency password when a customer has for some reason lost a valid ELC (and potentially the merchandise), but has been previously authorized to use the merchandise. Emergency passwords are particularly useful in a scenario where the customer is unable to reach the supplier of the merchandise using available contact information. For example, if the customer's hard disk is destroyed during a weekend, it is useful to be able to re-generate a valid ELC and potentially re-download the merchandise to allow the customer to continue to utilize an already purchased product.

Detailed Description Text (53):
The description thus far has primarily referred to use of the components of the client portion of the secure digital commerce system by a virtual store. One skilled in the art will recognize that many alternative configurations are possible. For example, a standalone online purchasing application can be used to execute the components of the DCS client to communicate directly to a licensing and purchasing broker to request and receive electronic licensing certificates. In addition, one skilled in the art will recognize that the separate components of the DCS client and the DCS server enable each component to be separately replaceable and separately customized. For example, to generate a customized virtual store, a specialized user interface for licensing and purchasing merchandise can be generated and stored as the user interface component (e.g., SAFEUI.dll 508 in FIG. 5) on the customer computer system. Further, one skilled in the art will recognize that the licensing code incorporated into the encrypted content (or content player) can be replaced in its entirety and can be made supplier specific. In addition, the code used to generate ELCs from the password generation data repository can be

optimized to be supplier specific. Further, all of the functions of the DCS server can be provided as licensing and purchasing administrative functions (for example, via an applications programming interface) to enable content suppliers to furnish their own licensing and purchasing brokers.

Detailed Description Text (59):
Although specific embodiments of, and examples for, the present invention are described herein for illustrative purposes, it is not intended that the invention be limited to these embodiments. Equivalent methods, structures, processes, steps, and other modifications within the spirit of the invention fall within the scope of the invention. For example, the teachings provided herein of the present invention can be applied to other client/server architectures, not necessarily the exemplary Internet based, HTTP model described above. These and other changes may be made to the invention in light of the above detailed description. Accordingly, the invention is not limited by the disclosure, but instead the scope of the present invention is to be determined by the following claims.

Current US Cross Reference Classification (1):
705/26

CLAIMS:

1. A computer system for conducting electronic commerce, including:

a store computer that receives requests for electronic data from a client computer and that, in response to receiving the request, sends to the client computer a download component that coordinates the download of the electronic data;

a supplier computer that receives a request from the download component of the client computer to download the electronic data and that, in response to receiving the request, sends the electronic data and a licensing component to the client computer, the licensing component for coordinating the licensing of the electronic data; and

a licensing computer that receives a request from the licensing component of the client computer to license electronic data and that, in response to receiving the request, determines whether access to the electronic data is to be allowed at the client computer, and when access is allowed, sends a notification that access is allowed to the client computer.

5. The system of claim 1 wherein the store computer, the supplier computer, and the licensing computer are separate web servers.

7. A method in a computer system for conducting electronic commerce, including:

requesting a first web server to order electronic data;

receiving in response to the request a download component for coordinating the download of the electronic data; and

under control of the download component, downloading from a second web server the electronic data.

8. The method of claim 7 wherein the download component also downloads a licensing component and including:

under control of the licensing component, requesting and receiving from a third web server a license for using the electronic data; and

using the electronic data in accordance with the received license.

9. The method of claim 8 including:

under control of a payment component, authorizing payment for the electronic data.

11. A method in a store computer for coordinating electronic commerce, the method including:

receiving from a client computer a request to purchase electronic data; and

in response to receiving the request, sending to the client computer a download component, the download component for coordinating the download of the electronic data from a supplier computer to the client computer, the supplier computer for downloading to the client computer the electronic data when requested by the download component.

13. The method of claim 11 wherein the store computer, the client computer, and the supplier computer communicate via the Internet.

Previous Doc Next Doc Go to Doc#

L9: Entry 9 of 13 File: USPT Oct 19, 1999

DOCUMENT-IDENTIFIER: US 5970475 A
TITLE: Electronic procurement system and method for trading partners

Abstract Text (1):
An Electronic Commerce system enables corporate purchasers and suppliers to electronically transact for the purchase and supply of goods/services. The system includes three major hardware and software components: buyer, supplier and bank/administration. To enable suppliers to supply goods and services online and process electronic orders, several software components are used for operating a supplier processor server and a supplier catalog server. To enable corporate purchasers to purchase products and services online, preferably over the Internet, from suppliers, software is used for operating a customer server to which multiple users may log-on and access the supplier server. An Automated Clearing House (ACH) server may be used to interface with a bank's (ACH) systems. A service bureau that supplies the hardware and/or software components and assists to administer the system includes a transaction counter, which records transactions and charges the buyers and/or suppliers based on the number of purchase orders and/or invoices issued.

Application Filing Date (1):
19971010

DATE ISSUED (1):
19991019

Brief Summary Text (3):
The present invention generally relates to electronic commerce systems and, more specifically, to a procurement system and method for trading partners which enables a plurality of users within a purchasing organization to procure goods/services from pre-arranged suppliers, consistent with the level of authorization given to each user and enables automated payments to the supplier by a bank after the goods/services have been delivered.

Brief Summary Text (6):
Ordering of non-production goods in high volumes, such as office supplies and desktop hardware, can be a time-consuming and expensive process for suppliers as well. For example, suppliers have to be increasingly competitive in today's market as their customers are constantly seeking immediate turn-around on orders and better overall customer service. However, suppliers find that the process of phone or paper purchase orders is costly because of the administration associated with order processing, can cause delays in order fulfillment, and is prone to errors. Additionally, paper-based marketing in the form of catalogs and price lists is expensive and makes it difficult to keep customers up-to-date with the latest product availability and pricing.

Brief Summary Text (7):
Ultimately, all these factors impact buyers through higher prices or poor service. Buyers have to implement time-consuming processes to prevent purchases that exceed employee limits. As a result, the cost of processing requisitions and purchase orders often exceeds the value of the goods being purchased. Buying organizations

also find it difficult to prevent employees from purchasing from non-preferred suppliers and thus do not get the advantage of negotiated prices. This adds to buyers' costs and reduces business for their preferred suppliers.

Brief Summary Text (11):
One example of an on-line system for processing business transactions is disclosed in U.S. Pat. No. 4,799,156 for an Interactive Market Management System. The system discloses a plurality of buyers and a plurality of sellers which can be linked to each other by means of an interactive market management system (IMMS) for interactive communications. Each of the participating entities which is a subscriber to the system must always operate through the IMMS, which serves as a focal point or hub through which all transactions must be funneled. The patent does not address the need or ability of individuals within an organization to be provided with different levels of authorization so that different users within the same organization or "buyer" can access different types and/or spend different amounts on goods and/or services.

Brief Summary Text (14):
In U.S. Pat. No. 5,592,378, a computerized order entry system and method is disclosed which includes a plurality of servers, data entry devices, back-end systems and data bases. The computer order entry system is intended to permit placement of orders by capturing order information and storing the order information through the data capture mechanism. This is accomplished by a sequence of steps of multiple search categories. The patent does not address the ready accessibility and ease of use by many employees within an organization to requisition goods/services from a pre-arranged trading partner or multiple partners.

Brief Summary Text (18):
It is still another object of the present invention to provide an electronic procurement system and method, as suggested in the previous objects, which is particularly suitable with buying organizations having a large number of employees each of which has well-defined authorizations for the purchase of goods/services in order to control such purchases and prevent abuses from within the organization.

Brief Summary Text (19):
It is yet another object of the present invention to provide an electronic procurement system, as suggested in the previous objects, which enables each user within a purchasing organization to use an Intranet connection to access the organization's Intranet Server as a means for accessing the supplier's server via an Internet connection by using an Internet browser.

Brief Summary Text (23):
In order to achieve the above objects, as well as others which will become apparent hereinafter, an electronic commerce system for procuring goods/services by a plurality of users within an organization, according to the invention, comprises a plurality of terminals. A customer server is connectable to each of said terminals and includes log-on means for providing access to a user by means of one of said terminals only if the user can be properly authenticated by the customer server. A supplier system is used which includes a supplier catalog server for storing data representing a supplier catalog of goods/services that are available for purchase by an authorized user in the customer organization and a supplier processor server for processing orders received from the authorized user within the customer organization. The supplier catalog server and the supplier processor server may be combined into one server. Said supplier system is directly accessible by said customer server through an Internet connection. Security means is provided within said servers which limit transactions to entities that have pre-arranged relationships for displaying supplier catalog information to an authorized user within the customer organization for issuing a purchase order by the user to said supplier system. A bank server may be used that is accessible by said customer

server through an Internet connection. Payments to the supplier by the customer organization may optionally be made through said bank server after the goods/services have been delivered to the user and an invoice has been issued to the customer organization.

Brief Summary Text (24):
Each user is preferably assigned an organization user profile which specifies a level of authorization for approval of the acquisition of goods/services from a pre-determined supplier. Said user terminals include means for displaying products/services available for acquisition from the at least one supplier and is consistent with the user's level of authorization for the acquisition of goods/services from said supplier. The supplier system includes a catalog and an order processor, said catalog containing information regarding all of the suppliers' goods made available to the customer organization, including pricing, discounts, availability, delivery information, etc., based on the organization's profile submitted to the supplier and negotiated agreement between the partners. A communication link is provided for selectively accessing, for viewing and downloading by a user, information from the supplier's catalog to the user's terminal consistent with the user's authorization level. Said customer and supplier systems are programmed to establish a cryptographically secure session for ordering and filling orders for goods, by means of said order processor from said supplier only when an authorized user seeks to acquire one or more products which the user is authorized to purchase.

Drawing Description Text (10):
FIG. 8 is a block diagram illustrating ACH security implementation between the procurement system and the bank server;

Drawing Description Text (11):
FIG. 9 is a flow chart for initial log-on by a user at a terminal of the buyer organization to gain access to the "main menu" on the customer server;

Drawing Description Text (13):
FIG. 11 is a flow chart representing the options/steps permitted by the customer server to a user at the buyer location after the "purchase" option has been selected from the "Main Menu" in FIG. 10;

Drawing Description Text (14):
FIG. 12 is a flow chart representing the options/steps permitted by the customer server to a user at the buyer location after a decision has been made to create or edit a template after the "Select A Template" option has been selected in FIG. 11;

Drawing Description Text (15):
FIG. 13 is a box diagram representing the "Administrative Main Menu" at the user terminal after the "Administration" option has been selected by a user from the Main Menu shown in FIG. 10, when the user is authorized to access the administration features of the customer server;

Drawing Description Text (16):
FIG. 14 is a flow chart representing the options/steps permitted by the customer server to a user at the buyer location after a decision has been made to access the "Supplier Management" module shown in FIG. 13 to set up and maintain supplies;

Drawing Description Text (17):
FIG. 15 is a flow chart representing the options/steps permitted by the customer server to a user at the buyer location after a decision has been made to access the "Supplier Management" module shown in FIG. 13 to set up and maintain supplier groups;

Drawing Description Text (18):

FIG. 16 is a flow chart representing the options/steps permitted by the customer server to a user at the buyer location after a decision has been made to access the "supplier management" module shown in FIG. 13 to specify which supplier employees can purchase from;

Drawing Description Text (19):
FIG. 17 is a flow chart representing the options/steps permitted by the customer server to a user at the buyer location after a decision has been made to access the "Employee:" module shown in FIG. 13 to use the "Main Employee Manager";

Drawing Description Text (20):
FIG. 18 is a flow chart representing the options/steps permitted by the customer server to a user at the buyer location to create and maintain employee information files or profiles;

Drawing Description Text (21):
FIG. 19 is a flow chart representing the options/steps permitted by the customer server to a user at the buyer location to create and maintain employee groups;

Drawing Description Text (22):
FIG. 20 is a flow chart representing the options/steps permitted by the customer server to a user at the buyer location after a decision has been made to access the "Authority Setup" option on the Administration Main Menu in FIG. 13;

Drawing Description Text (23):
FIG. 21 is a flow chart representing the options/steps permitted by the customer server to a user at the buyer location after a decision has been made to set-up and maintain bank accounts from the main accounting module in FIG. 20;

Drawing Description Text (24):
FIG. 22 is a flow chart representing the options/steps permitted by the customer server to a user at the buyer location after a decision has been made to set up and maintain supplier payments from the main accounting module in FIG. 20;

Drawing Description Text (25):
FIG. 23 is a flow chart representing the options/steps permitted by the customer server to a user at the buyer location after a decision has been made to access the "Payment" module from the Administration Main Menu in FIG. 13;

Drawing Description Text (26):
FIG. 24 is a flow chart representing the options/steps permitted by the customer server to a user at the buyer location after a decision has been made to access the "Reports" module from the Administration Main Menu in FIG. 13; and

Detailed Description Text (3):
The Electronic Commerce (EC) system in accordance to the present invention is an electronic purchasing/invoicing/settlements system built from a combination of off-the-shelf hardware and software packages and custom software. It is intended to reduce costs by allowing authorized end users to directly interact with suppliers, rather than through centralized purchasing agents.

Detailed Description Text (5):
Referring to FIG. 2, the generalized functional and structural components linking the Buyer organization or Customer 12 and the Seller or Supplier 16 of FIG. 1 are illustrated. In the preferred embodiment, a user 24, usually an employee of the buyer organization or customer 12, uses one of a plurality of terminals 26 to access the system. After entering the user's network as a valid user by the customer's Intranet server 30. The terminals 26 may be linked to the customer server 34 in any conventional way. The user 24 can access the EC System by entering user's "application" log-on password and, being authenticated. The user is then

able to enter the EC System.

Detailed Description Text (6):
Typically, once identified as a valid user by the customer server 34 requisite levels of security, in the form of firewalls 32, 36, must initially be penetrated on both sides of the customer server 34 to gain Internet access. Using a browser, such as NETSCAPE.RTM. or MICROSOFT.RTM. INTERNET EXPLORER.RTM., the user 24 can access the seller or supplier 16 by penetrating another level of network security, in the form of a firewall 38, and being authenticated by the supplier 16.

Detailed Description Text (7):
The supplier 16 has a supplier processor server 40 and a catalog server 42. The supplier may also have a legacy system 44, in which case the supplier may also require a gateway (not shown) to the legacy system. The user 24 may access the catalog server 42, in whole or in part. The level of access is established by an administrator 58 (FIG. 5) who defines the user profile for each user in the buyer organization 12. Such profile establishes what part of the catalog server 42 the user may view, the user's spending limits, the nature of the goods/services that the user may procure, the nature of the administrative options that the user may access and execute, etc.

Detailed Description Text (9):
a) Customer or Purchaser system 12 that includes at least one terminal 26, a Customer Server 34 (at a buyer's site running on buyer hardware). The customer system 12 may also include a purchaser legacy system 46, if the customer already has existing hardware and/or data that can be used with the customer server 34.

Detailed Description Text (10):
b) Supplier Processor Server 40 (at a vendor site running on vendor hardware). The Supplier Processor Server 40 may access Supplier's Legacy System or Servers 44, 44'.

Detailed Description Text (11):
c) Supplier Web Catalog Server 42 (running at a vendor site running on vendor hardware). The supplier legacy system 44 may include legacy catalog 44'.

Detailed Description Text (12):
d) Certificate Authority Server 54 (running off-line).

Detailed Description Text (13):
e) Bank Server 18, which may be accessed directly or by means of Automated Clearing House (ACH) Gateway 50;

Detailed Description Text (14):
f) Transaction Counter Server 52 running at a Service Bureau (SB).

Detailed Description Text (15):
The supplier catalog server 42, 44' may not be physically located near the supplier order processing system 40, 44, and these various components can be linked in any conventional way to access each other.

Detailed Description Text (16):
In addition, a Java-enabled Web browser is required, running at buyer sites on buyer-supplied PC class computers and communicating with the buyer's Customer Server 34, typically over the buyer's Intranet/LAN L.sub.1. Buyer and supplier hardware running these servers use MICROSOFT WINDOWS NT SERVER 4.0.RTM. and several other MICROSOFT.RTM. products in support of the electronic commerce (EC) system.

Detailed Description Text (22):
More specifically, referring to FIG. 2, the EC System is generally designated by

the reference numeral 10 (FIG. 1). The Buyer or Customer System 12 includes a plurality of user terminals 26. Although only one terminal is shown, hundreds or thousands of user terminals may be used in a large company, either at one location or at a plurality of distributed or remote locations or facilities. One feature of the invention is the connection of the user terminal(s) 26 to a Customer Server 34. Preferably, the user terminal(s) 26 are connected to the Customer Server 34 by means of an Intranet link L.sub.1. The Customer Server 34 may be connected to a Purchaser Legacy Data Base 46.

Detailed Description Text (23):
Referring to FIG. 3, the Customer Server 34 is connectable to the Supplier Order Processor 40 by means of the Internet (Links L.sub.2, L.sub.3) and the Supplier Catalog Server 42 by means of Internet Link L.sub.3 ' when the components 40, 42 are nor linked to each other but must be separately accessed. The Supplier System 16 includes a Supplier Order Processor 40 connected to the Supply Legacy System 44, which represents a supplier's original hardware and/or data base or archival records. The Supplier Catalog Server 42 may be connectable to a Legacy Catalog 44' which represents a supplier's original electronic catalog.

Detailed Description Text (24):
The Customer Server 46 is optionally connectable to the Bank Server 18, directly or by means of an ACH Gateway 50. The method chosen to access the Bank Server will be determined by the format of the data that will be accepted by the Bank. Where the Bank can accept instructions directly from the customer server 34, the ACH Gateway 50 may be omitted. When used, the ACH Gateway 50 may be located at a service bureau (SB) 48, where the counter 52 may also be located.

Detailed Description Text (25):
FIG. 4 illustrates the general connectivities between the primary blocks shown in FIG. 3, and highlights the direct connectivity between the buyer 12 and the supplier(s) 16, without any intermediary agents. Similarly, there is direct connectivity between the buyer 12 and the bank 18, except when the bank requires translation of data to a format consistent with it's own server, in which case the ACH Gateway 50 must be used. The counter 52 at the service bureau only monitors and counts transactions based on purchase orders sent by the customer server 12 to the supplier(s). This allows the service bureau to be compensated on a per transaction basis. However, clearly, other payment arrangements can be made to compensate the service bureau, such as flat annual or periodic payments, in which case the counter 52 may be omitted in some instances.

Detailed Description Text (26):
Referring to FIG. 5, an Administrator 58, can set up the parameters of the user hierarchy within the procurement system 12 to establish which users 24 can access which portions the Supplier's Catalog 42, 44' and other administrative functions. In this way, each user 24 must be authenticated for level of access, for requisition and/or administration. The bank server 18 is registered directly with the customer using FIMAS authentication.

Detailed Description Text (27):
The Procurement System 12 serves to authorize users 24 to display catalogs, search and select goods, order inquiries and request quotations. The Supplier System 16 processes orders, queries, and sends invoices. The Bank Server 18 processes ACH settlements and the Counter 52 counts transactions. Aside from registering the Procurement System 12, as aforementioned, the Administrator 58 also establishes the authorization level for each user 24 when using one of the terminals 26 to define what portion of the Supplier catalog 40, 42 each user can access, the nature of the goods that the user can order and/or the spending limit that each user has for purchasing the designated goods/services.

Detailed Description Text (28):

FIG. 5 illustrates the basic transactions that the EC System 10 can execute. The Procurement System 12 sends purchase orders 64 to the Supplier System 16, while the Supplier System 16 issues sales invoices 62 to the Procurement System 12. The goods/services ordered by the user 12 are supplied, at 64 to the user. The purchase order information is stored, or may be retrieved, at 66 from the Supplier's Legacy System 44, 44'. After a shipment has been made to the user, the user can select how to pay the invoice, e.g., by creditor purchase card, check, Legacy System, or ACH. If ACH is selected, a transaction event 68 is transmitted to the bank's transaction processor 50 to input into the Bank Server 18 the nature of the transaction and/or the payment(s) that need to be made to the Supplier. The Counter 52 counts the number of transactions so that the service bureau 48 can be compensated for it's payment, clearance and settlement services. General system requirements are global components that provide common functionality. For example, all functional components have potential requirements for error handling; but for the sake of uniformity and efficiency, these requirements are moved from their respective functional components into the common error handling functional component category.

Detailed Description Text (30):
1. Security--for encryption, decryption and authentication,

Detailed Description Text (38):
Users 24 can have access to different areas of functionality within the Procurement System 12, depending on the tasks they need to carry out for their jobs. The main areas of functionality, established within the Customer Server 46 are:

Detailed Description Text (45):
Software is provided for the suppliers to enable them to set up an online catalog, specify customer-specific pricing profiles and to process purchase orders received from corporate purchasers.

Detailed Description Text (46):
The Supplier System 16 has two major components. The supplier catalog server 42 that provides all catalog format, maintenance and browsing capabilities. The Supplier Order Processor Server 40 provides connectivity to legacy supply systems 44 for order verification and notification.

Detailed Description Text (48):
The EC system includes a Web server 72 for linking the terminals 26 to the customer server 34 by Intranet link L.sub.1. A secure socket layer (SSL) is provided by the Internet information server (IIS), provided by the MICROSOFT.RTM. Corporation.

Detailed Description Text (49):
The customer server 34 is advantageously provided with a number of software modules, including "Accounting" 74 for monitoring accounts and generating reports; "System Management" 76 for setting up the system and maintaining its performance; "Purchase Order" 78 for maintaining and performing purchase transactions; "Event Handler" 80 for handling events and errors; "Reporting" 82 for generating system reports. MICROSOFT.RTM. SQL purchasing data base server 84 is used which can be accessed by ODBC or ASP. A "Secure Log-On Module" 86 is used to authenticate users 24, by checking passwords, smartcards or tokens. Where the buyer chooses to make payments using the ACH procedure, the instructions may be sent directly to the bank 18 or through an ACH Gateway 50 by means of an account settlement module 88 to settle transactions. Where the buyer or customer has a legacy system 46, previously used for accounting, HR, MIS or EIS, the customer server 46 preferably has a "Legacy Interface" 90, using electronic data interchange (EDI) and/or custom translator. A "Transaction Control" module 92 monitors order transactions (purchase orders) by transmitting this information to the counter server 52 (FIGS. 3-5), preferably over an Internet connection, to allow the SB 48 to record and manage transaction details.

Detailed Description Text (50):
"Security Layers" 94 are used in both the buyer and supplier systems 12, 16 to provide authorization and encryption/decryption of critical communications. "Synchronous Communication" modules 96 enable data and message transfers, over the Internet 14, between the buyer and supplier systems.

Detailed Description Text (51):
The supplier catalog server 42 includes a "Catalog SQL Server" 98, similar to the purchasing catalog 84, for maintaining a catalog data base. The "Active Server Pages" 72 include a "Line Item Processor" 100 for retrieving and sending line items to the customer server 34. A Data Replication Unit 102 may be used to receive data from the legacy catalog 44'.

Detailed Description Text (52):
The supplier processor server 40 includes an "Order Processor" 104 which processes and controls the requisitions or purchase orders received from a customer user, while the "supplier" module 106 provides temporary storage for data generated during such processing. Where the supplier has a legacy system 44, a suitable "Legacy Interface" 108 is used to selectively communicate with the suppliers legacy system 44 and/or the legacy catalog 44'.

Detailed Description Text (54):
The EC system core functionality is implemented using many Microsoft's component object model (COM) which are derived from the Active Template Library (ATL) tools. Each COM object encapsulates a specific functional role and provides a function interface that can be accessed by other COM objects or COM-enabled processes (such as Active Server Pages 70 in FIG. 6). A layer of abstraction consists of several COM components. It is these components that will be used in Active Server Pages (ASP) 70. This allows changing the implementation of any underlying component--to use a different database table, for example--without affecting any high-level program or business logic encapsulated in either ASP or COM objects ASP is using.

Detailed Description Text (56):
The customer server has a distinct communications module 96. This enables the customer server 34 to transmit documents and receive documents over a variety of protocols. The transport method currently used is a direct server to server Internet Protocol (IP) socket connection. This connection uses the standard HTTP protocol as used by Internet Web servers such as IIS and NETSCAPE.RTM. Commerce Server. The purchase order document that is sent from the customer server to a supplier is formatted to adhere to the OBI (Open Buying on the Internet) specification. All other documents are sent as per standard ANSI X.12 EDI specifications. The method for instigating a transfer of a document is via an HTTP POST operation to a waiting CGI program at the supplier site. The CGI program will decode the object and place the document as a text file ready for integration into a supplier order entry system.

Detailed Description Text (64):
2. Connection access to Procurement Server.

Detailed Description Text (65):
Customer Server Requirements

Detailed Description Text (67):
1. Windows NT (server/workstation) 4.0 operating system.

Detailed Description Text (69):
Microsoft Windows NT Server v4.0 operating system

Detailed Description Text (70):

Internet Information Server v3.0 with Active Server Pages extension

Detailed Description Text (71):
SQL Server v6.5 (recommended, other ODBC compliant database systems may be used).

Detailed Description Text (72):
CryptoAPI v2.0 component of NT Server v4.0

Detailed Description Text (74):
Procurement Server

Detailed Description Text (78):
1. Windows NT (server/workstation) 4.0 operating system.

Detailed Description Text (80):
1. Windows NT Server V4.0 operating system

Detailed Description Text (81):
2. Microsoft Internet Information Server 3.0

Detailed Description Text (82):
3. Microsoft SQL Server 6.5 (recommended, other ODBC-enabled Database may be used)

Detailed Description Text (88):
1. Windows NT (server/workstation) 4.0 operating system.

Detailed Description Text (91):
Microsoft Windows NT Server v4.0 operating system

Detailed Description Text (92):
SQL Server v6.5

Detailed Description Text (93):
CryptoAPI v2.0 component of NT Server v4.0

Detailed Description Text (94):
Payment Clearing Server

Detailed Description Text (95):
Payment Clearing Server System Requirements

Detailed Description Text (97):
1. Windows NT (server/workstation) 4.0 operating system.

Detailed Description Text (100):
All the Servers currently use Microsoft's SQL Server. All are completely ODBC compliant for ease of migration to other ODBC databases, e.g., Oracle.

Detailed Description Text (101):
The Purchase Interface and Administrative Interfaces run as Java applet suites within a Web browser (such as Netscape or Internet Explorer) and communicates with a buyer's Customer Server 34 using HTML via Microsoft's Internet Information Server (IIS) with the ASP extension. They are the user's point of contact with the entire System. The User Interface is important for that reason. Because users must find the procurement system 12 friendly, intuitive, and easy to use, committed to improvements in the User Interface (UI) as users suggest improvements or as additional features are incorporated in the future.

Detailed Description Text (102):
Because of Bank security concerns, at least one Service Bureau (SB) 48 is

preferably provided. Each SB will physically consist of assorted network components implementing Internet Firewalls, the Payment Clearing Server functionality (i.e., transaction counting and optional ACH translation). An approved certificate authority 54 will be provide certificates for system security.

Detailed Description Text (107):
From a security viewpoint, the EC system can best be described as an extranet (internet closed environment) server-to-server communication architecture. One customer server 34 communicates using EDI messages with several supplier or supplier servers 40, 42.

Detailed Description Text (108):
The Customer Server 34, the Supplier Server 40, 42, Certificate Authority 54, and Bank Servers 18 are software products which run on Windows NT Server v4.0 capable hardware. At Customer sites, it is the responsibility of Customer system administration personnel to configure their equipment, and to ensure that system installation and configuration meet internal customer standards for safe computing. For example, a system administration will probably wish to consider such issues as firewall policies, log-on security behind the firewall, hardware and network redundancy, backups, disaster recovery, physical access controls, etc. It is will recommended that every customer implement effective and properly configured firewalls between all machines running customer server software and the open Internet.

Detailed Description Text (109):
Buyers use the Customer Server 34 to communicate over the open Internet 14 with any of several Suppliers using EDI, ANSI or EDIFACT standard messages to Suppliers using a Supplier Servers 40, 42. The System utilizes open architecture design conforming with known standards, such as the OBI (Open Buying on the Internet), but from a security viewpoint is a closed business system because buyers and sellers are already established and trusted trading partners before using the EC System. The buyer's Customer Server 34 knows the identity of all Supplier Servers 40, 42 (and vice-versa), which obviates several security and business relationship problems which might be encountered if unknown or unregistered Suppliers were allowed to connect to the system.

Detailed Description Text (110):
It is a closed communication system because traffic is permitted only among pre-established and specially configured computers. Customer (and other) computer equipment not running system software cannot establish a communication link. No Internet services will be available (e.g., finger, mail, ftp, . . .) from systems running Server software except for an HTTP daemon/per Server to service users operating at client PCs. HTTP is planned as the transport protocol of choice (for security reasons) when it is fully supported by an NT Server. Before any messages whatsoever can be exchanged between servers, each side must be authenticated for the other using authentication certificates signed by the central Certification Authority Server 54 (FIGS. 3 and 7) at a Service Bureau 100. Every message sent on the open Internet by any system Server is encrypted using RSA public/private key techniques, even prior to successful authentication. All messages are in limited, standard forms (e.g., EDI forms, PKCS messages, etc). No other communication over the open Internet 14 to, or from, a system Server is permitted or attempted.

Detailed Description Text (111):
All System Servers run under the Microsoft Windows NT Server v4.0 operating system w/Service Pack 3 and use NT file system and other security features. All System Servers must be physically protected by placement in secure locations. A properly configured (and located) NT Server 4.0 with Service Pack 2 or better meets U.S. Government/NSA C2 security as a server. It also meets the UK Information Technology Security Evaluation and Certification standard at the FC2/E3 level.

Detailed Description Text (112):
Users 24 typically connect to their Customer Server 34 across their corporate Intranet/LAN (L.sub.1 in FIGS. 2, 6) using off-the-shelf Java-enabled browsers 111 (at least Netscape navigator 2.02 or Microsoft Internet Explorer 3.0--at Java v1.0.2). User names and passwords are encrypted during log-on using SSL services.

Detailed Description Text (113):
No additional software need be installed on a user workstation 28. In particular, no ActiveX or similar system-executable processes are ever downloaded to or executed on the user workstation/PC 28 by any part of the System; this was a deliberate choice to increase security. Java applets are used as visual enhancements for the browser running the Administrative Interface to the Customer Server 34; the applets do not, and cannot by design, have access to any part of the PC running the Purchase or Administrative Interface for buyer users. Applets are downloaded into the user's browser only over the Customer Intranet 28 and only from an Customer Server 34; at no time are Java applets ever loaded from the open Internet 14. The Purchase and Administrative Interfaces to the Customer Server 34 operate through server-side processing and dynamic server-side HTML page generation using Microsoft's Internet Information Server IIS 72 with the Active Server Pages extension 70. All user access to the Customer Server 34 is exclusively through HTTP, which provides a high degree of server encapsulation and security. Microsoft's IIS 90 acts as the web server for user browsers and serves as the gateway to the Customer Server 34 itself.

Detailed Description Text (114):
Server-to-Server communication security ultimately relies on public key cryptographic techniques as implemented in the RSADSI TIPEM library. The TIPEM library implements RSADSI's Public-Key Cryptography Standards. Referring to FIG. 7, the server 110 at the certification authority 54 function will be performed by Motorola's CipherNet application, which will run off-line. This certificate authority application will be managed and run by Service Bureau hardware at the location 54. Each System Server includes its own authentication certificate [C1] in all messages it sends; a Server-resident copy of Motorola's CipherNet Toolkit makes calls as needed to the TIPEM library to produce a PKCS package which contains a copy of its authentication certificate digitally signed by the SB Certification Authority. Only Servers need authentication certificates since Internet communication within the SB System is exclusively server-to-server; users 24 are authenticated by the Customer Server 34 using username/password authentication. Increased security by utilizing local certificates is an option the customer can choose for some or each user. Messages between Servers contain signed and encrypted EDI standard forms plus the sending server's certificate in a PKCS #7 compliant package. This allows the receiving server to be certain of (i.e., to authenticate) the sending Server's identity since only the sending Server's public key will successfully decrypt the signed digest, and the message digest itself prevents error, garble, and effectively all tampering with the contents of the message. Microsoft's COM objects is the remote procedure call mechanism and uses TCP/IP as its transport protocol.

Detailed Description Text (115):
The Customer Server 34 maintains a list of all valid supplier or vendor server certificates and user 20 profiles as well as usernames and passwords.

Detailed Description Text (116):
The private key used by each Customer Server 34 to authenticate itself to other Servers, never leaves the Server. Furthermore, even in memory, it exists only transiently in unencrypted form. It is secured on the Server with several layers of security measures. First, it is stored in the NT Server's Registry (which is kept on disk) in CipherNet encrypted form. The key needed to decrypt it is kept (also encrypted), in a large otherwise pseudo-random, file. Extraction and decryption of the secondary key requires access to assorted parts of particular system

information and to a large block of system-generated pseudo-random data.

Detailed Description Text (117):
A Server's RSA public/private key pair comes into existence when a request to produce them is made of the local copy of the CipherNet Toolkit which in turn calls on the local copy of the TIPEM library to actually do the work. Referring to FIG. 7, the certificate request (as a PKCS #7 message in a file on a disk 112) will be physically brought to the Service Bureau 54 via the Customer's choice of secure transport 56. The Service Bureau-managed Certification Authority server 110 will digitally sign it, producing an identity certificate for that Server on disk 112' containing the Server's public key, among other data. The request to produce a Server's RSA key pair (which it must have to communicate any other Server) is made at the Customer's direction and only by the receiving Customer. The resulting digitally signed authentication certificate for the Server's public key will be installed on a Customer's system at the Customer's site by certificate authority personnel under Customer supervision, using a Customer-chosen and Customer-entered password. At no time will a signed Server authentication certificate from the certification server, not already installed on Customer hardware, exist outside the Customer Authority except in the care of an Customer Authority employee.

Detailed Description Text (119):
Server-to-server security uses public/private key techniques based on the RSADSI TIPEM library. Motorola's CipherNet Certifier application will be run off-line on a stand-alone NT workstation 110 providing signed authentication certificates for all System software at all sites. This certificate authority will be managed and run by an approved Customer Authority on it's hardware. Each server uses a local copy of Motorola's CipherNet Toolkit to generate signed and encrypted messages in PKCS #7 format for exchange with other Servers; these always contain a copy of the Certification-Authority-signed certificate for that Server. Messages contain EDI standard forms, plus the sending Server's signed certificate. They are readable only by the intended recipient. This allows the receiving Server to authenticate the sending Server, by successfully decrypting the sender's certificate and then the signature of the Certification Authority Server. Users do not have certificates since communication is Server to Server; users are authenticated by the Customer Server 34 using username/password authentication.

Detailed Description Text (121):
In order for the EC system to begin standard operation, each EC server must have its own certificate containing its public key, a private key, the server's information (AVA, DER), and the signed certificate of CipherNet.

Detailed Description Text (122):
Note that user or customer browsers 11 do not require their own certificates. Certificates are not necessary (but optional for higher security requirements) at the user side since the Customer Server 34 authorizes users by username/password authorization. Server to server communication use server certificates to authenticate servers.

Detailed Description Text (137):
13. In Set X12 Security Options accept default settings. All messages are secured using both authentication and encryption, No filtering, Functional Acknowledgements not secured.

Detailed Description Text (144):
20. Authentication Character (if using EBCDIC).

Detailed Description Text (146):
22. Accept default encryption and authentication key names.

Detailed Description Text (147):

23. Assuming manual key management; from the Manual Key management screen, for each key (i.e. authentication key and then encryption key) enter date and time from which key is to be valid, then enter each of the three 16 hexadecimal character strings which are mathematically combined to generate the key.

Detailed Description Text (157):
Administrative functions allow the Procurement System 12 administrator 58 to add users 24 and their profiles to the Customer Server 34. Access to these administrative functions is accomplished through a standard user web browser 111. Special administrative functionality is enabled when it is determined by the Procuring system that the user that is trying to log on has special administrative privileges. Administrators access to the Customer Server 34 through a username/password combination that is identical to the normal user access procedure. The user profile allows the administrator to restrict a user's catalog view to purchasing profiles to be edited and sets spending and types limits on purchase order items. User profile information is located on a SQL Server database 54 on the Customer server 34, which is secured physically from direct casual access.

Detailed Description Text (174):
Each Procurement and Supplier server must have its own certificate. Furthermore, each certificate must be signed (certified) by CipherNet. In order to obtain a signed certificate from CipherNet, each server must create a certificate request. Referring to FIG. 7, the format of the request is a PKCS#7. The process that creates this request is CERTREQ.EXE 114, the Certificate Request Wizard.

Detailed Description Text (185):
The wizard 114 also writes the private key to the SQL server database 84, 98 (FIGS. 5, 7), which is later used by the security COM component during program operation. Although the public key is also defined, it is not yet saved to database. This is because the public key is passed in the certificate request, and eventually returns embedded in the signed certificate. The signed certificate, and consequently the public key, is stored by CERTPROC.EXE in the Certificate Processor 116.

Detailed Description Text (187):
There is a Customer Server 34 at the Buyer's location. This system 12 is responsible for encrypting and sending EDI requests to Suppliers, for notifying the Bank Server 18 of billable transactions, and of sending messages to the bank ACH authorizing payment of invoices. In the last case, the Payment Clearing Server takes the message and translates the encrypted message from calls to SMTP mail and routes it to the Bank's Server 18 which translates the SMTP mail back into EDI format and normal Bank ACH processing continues. EDI forms 997 and 824 are generated and sent back to Bank Server, translates the SMTP mail back to COM calls and forwards it to the Customer Server 34. The Bank Server 18 decrypts the message and logs the transaction.

Detailed Description Text (188):
There are no manual processing procedures built into the Bank Server workflow. However, there are exception handling procedures that are handled manually. Such processes handle message routing exceptions, e.g., when no EDI 997 (i.e., functional acknowledgment) is received from the Bank's ACH. In such cases, the operator will be notified after a predetermined period of time and requested to re-send the message after checking with the bank. Due to the use of electronic mail as the transport vehicle to and from the bank's ACH, this procedure is in place for message failures.

Detailed Description Text (189):
A transaction or server counter 52 and billing module for the Service Bureau 48 is provided. Manual input occurs when a new Customer is set up and the transaction rates which have been agreed to by contract with the Customer are entered into the

Counter Server 52. The invoicing of Customers will be done automatically using information provided by the Server 52.

Detailed Description Text (190):
The Counter Server 52 produces customer billing information, based on the billing period entered into the customer profile, to automatically invoice the customer directly. This billing procedure can interface to the Service Bureau's general ledger accounting. Full audit trails and various reports on customer activity can be provided by the Bank Server 18 and Counter Server 52.

Detailed Description Text (192):
The SB 48 and Bank Server 18 can provide various reports on Customer transactions. Users can use a report writer to extract information from the database. Audit trails on both ACH and billing transactions can be kept by the Bank Server, so that all ACH transactions have a full event history. The Bank Server will also provide a user event log, which will include an entry for each instance in which a user alters information on the system. These alterations will be logged and available for audit.

Detailed Description Text (198):
4. User name and password sent to Customer Server 16 for authentication.

Detailed Description Text (199):
5. Authenticated user is presented with his purchasing profile details.

Detailed Description Text (200):
6. For each item selected, the line item detail is sent from the Supplier Catalog Server 24 to populate the purchase requisition.

Detailed Description Text (201):
7. User reviews the purchase requisition that is sent to Customer Server for processing.

Detailed Description Text (202):
8. The Customer Server 34 verifies the purchase requisition against the user's profile (including spend limit).

Detailed Description Text (206):
10. The EDI Purchase Order is encrypted using the public key from the supplier's certificate, signed using the Customer Server private key and placed in a PKCS #7 along with the Customer Server certificate. The encrypted/signed Purchase Order is sent to the Supplier.

Detailed Description Text (207):
11. The Supplier's Server 40 decrypts the purchase order with its private key, and verifies the signature by decrypting it with the public key contained in the buyer's certificate contained in the PKCS #7.

Detailed Description Text (208):
12. The Supplier Server 40 stores the buyer's public key for use in encrypting messages back to the buyer.

Detailed Description Text (210):
14. The Customer Server 34 triggers ACH payment (using FIMAS , EDI-820) mechanism.

Detailed Description Text (211):
Many of these functions use common security components to perform their tasks. For example, all EDI message transmissions between the Supplier and Customer Servers use a common encryption/decryption and authentication module.

Detailed Description Text (213):
IIS 72 uses standard NT security when it operates, which means that even though it is triggered by a remote process it still needs to log on to some NT account in order to operate. Since it is neither feasible nor practical to have an account for every client browser that connects (even if it knew them all!), ITS by default uses an anonymous account to log onto NT. However, IIS is configured during installation to disallow anonymous connections for security reasons. This is necessary to stop unauthorized Intranet client connections to the IIS. Only users that have registered with the system (added to the SQL database 84 by the Purchaser administrator 58) should gain access to the Customer Server 34 through IIS. But disallowing anonymous connections forces IIS to automatically prompt the user with a standard NT logon dialog (username/password). Since the user does not or should not have an NT account on the Customer server, a method to allow IIS to first-validate the user against the Customer Server SQL database 84, and, if found, second-allow IIS to log on using a `known` account. An information server API (ISAPI) extension obtains the username and password, validate these against the database, and logon using the known account. The known NT account would have been previously created by the Procurement Server administrator. Its name and password must match the contents of the ISAPI extension that works by substituting the user-entered name and password (which was used to validate against the database) with the known NT account name and password which is stored in the ISAPI extension. The password is not required to be known by anyone other than the administrator.

Detailed Description Text (214):
The username/password is encrypted when it is transported from the client browser to the server using SSL 2.0, which is inherently provided by using IIS 3.0 and a modern browser.

Detailed Description Text (215):
The user connects to the supplier's ITS site 70 using the Customer Server ITS, which obtains the supplier's URL from the Customer Server database 84 and connects to the supplier's IIS (Supplier catalog server 42). The Supplier catalog server 42 receives the Customer Server 34 certificate as a moniker in the URL during the initial connection. The Supplier Server can authenticate this certificate to confirm that a valid user is connecting to the catalog. Also, a buyer profile code is sent as a moniker to the Supplier Catalog Server 42. The Supplier Catalog Server 42 uses Active Server Pages 70 to dynamically create HTML catalog pages using data from its resident SQL Server database 98. By evaluating the buyer's profile, the Supplier Catalog Server 42 can custom display catalog information and pricing specific to the buyer or buyer's organization.

Detailed Description Text (216):
The Customer SQL Server database 84 contains a list of valid Supplier Server URLs and their certificates. The Customer Server 34 provides an administrative function that can download Supplier certificates at any time. These certificates are not validated at this point since any invalid certificates--such as certificates that haven't been signed by a common root--will be discovered immediately when the Customer Server 34 receives an EDI message from the Supplier Server 42. The Customer Server authenticates all EDI messages that are sent from the Supplier Server, which will have been encrypted and signed by the Supplier Server.

Detailed Description Text (221):
4. Message Authentication Key Size/Algorithm: 1024 bit RSA (Using Buyer's 1024 bit private key)

Detailed Description Text (223):
Referring to FIG. 8, the system will leverage the existing Bank's Templar Gateway when sending 820 EDI messages that are required by the Bank's ACH payment mechanism for account settlement. A mail client process will create a Templar-compliant MIME message that is sent to the Templar Gateway at the bank. The message is in the

bank-specified 820 EDI format. Bank's ACH security software will be used to manage FIMAS keys and MAC and encrypt outgoing 820 messages.

Detailed Description Text (226):
2. A component is triggered by the Customer Server system 12 to perform ACH payment. A bank-specific 820 EDI message is created.

Detailed Description Text (228):
4. The mail client application 122 packages the encrypted and MAC'd 820 EDI message into a MIME mail message and mails it to the bank 18.

Detailed Description Text (230):
6. The acknowledgement EDI message (997 or 824) is E-mailed by the bank 18 to the mail client application.

Detailed Description Text (231):
7. The mail client application passes the EDI message to the ACH COM component 96. The COM component 96 reconciles the acknowledgement EDI message against a list of outstanding 820 messages. If an 820 EDI message is not acknowledged within a user-defined amount of time, an administrative alert is sent.

Detailed Description Text (234):
Where applicable, EDI messages will be used to communicate information between applications forming part of the Electronic Commerce solution, a system that enables suppliers and corporate purchasers to supply and purchase goods and services electronically. Purchasing orientated EDI messages will be used to communicate purchasing information. Purchasing information sent will be constrained by the ANSI.X12 version and implementation standard chosen. Bank Financial EDI messages will be used to communicate settlement information between the system's servers. Settlement information sent will be constrained by the bank's ANSI.X12 implementation standard.

Detailed Description Text (266):
Referring to FIG. 9, the log on procedure by a user 24 for initial access to the system is illustrated. From the desktop or terminal 26, attempts to log on, at 124, 126 triggers a user's terminal to first be checked for network security at 125. Only if the network security user level has been satisfied does the desktop prompt the user to enter name and password at 127. Only if the proper IDs have been entered and logon security has been satisfied the Customer Server 34 authenticates the user 24, at 128 and set the user's privileges or predefined purchasing parameters and/or limitations, at 129. Once this has been established, the terminal 14 provides access to the Main Menu at 130, shown in FIG. 10.

Detailed Description Text (267):
The user 24 has a number of options in the Main Menu 131, including making a purchase (131a), administrative tasks (131b) (only if Customer Server has identified user as an administrator), review tasks waiting (131c), check order status (131d), review reports (131e), obtaining purchasing instructions (131f), changing the password (131g), or reviewing user feedback (131h). This Main Menu is illustrative and other options may be added and shown options may be deleted if a purchaser does not require one or more of the functions shown.

Detailed Description Text (268):
FIG. 11 illustrates the options when an authenticated user selects the "purchase" option 131a in FIG. 10. The user can create a new requisition request (132), repeat an old requisition request (133) or select a template (134). When selecting a template at 134 in FIG. 11, the user can also create a new template (134a) or edit an existing template (134b), as suggested in FIG. 12. In each case, the user is prompted to provide instructions or information. The steps are, in each instance described in the blocks. In each case, it will be noted, the user's request is

compared with the authorized limits for the user. If any of these limits or parameters are exceeded, the system interrupts the procedure and also the user if the user wishes to proceed with the request. If the answer is "no", the user may edit the request to bring it with the specified limits. If the answer is "yes", the requests sent to a supervisor's terminal for review. When a user is also an administrator, authorized to access management modules, the "Administration Main Menu" 131b can be accessed by electing the "Administration" option 131b in FIG. 10. The "Administration Main Menu" provides the administrator 58 with the following options: account wrapping 135, system functions 136, employees 137, accounting set-up 138, reports 131e, supplier management 140 and payments 141. Again, the options in the "Administration Main Menu" 131 in FIG. 10, modules may be added or deleted to serve the specific needs of a buyer or customer.

Detailed Description Text (276):
1. An item (product or service) is loaded from Supplier's Legacy Catalog 44' to the SQL Supplier Catalog Server 98 to make the item viewable to a customer user.

Detailed Description Text (280):
5. User sends requisition request to purchasing manager module in the customer server 34. This step is automatically effected when the item selected by the user exceeds the authority or purchasing parameters of the user as defined by the user's profile established by a customer administrator or manager.

Detailed Description Text (286):
11. Invoice and payment settlement request is forwarded to the Bank (ACH) Server.

Detailed Description Text (294):
1) Substantial cost savings by reducing administration involved in processing phone or paper orders; and eliminating the need for paper-based catalogs or price lists to keep customers up-to-date.

Detailed Description Text (303):
1) A certification authority which uniquely authenticates buyers and suppliers.

Detailed Description Text (304):
2) Public/private key technology to provide encryption/decryption, authentication, integrity and non-repudiation of all messages.

Detailed Description Text (308):
5) Only authorized requisitioners can log on to the system, using username and password technology.

Detailed Description Text (311):
1) Any customized catalog features, such as discounted prices for specific customers can be viewed only by those customers.

Detailed Description Text (313):
The system provides a total solution because it enables all parties involved in the procurement process to work together electronically through all phases of this process. Thus, for example, requisitioners can complete requisitions quickly and accurately by selecting goods from electronic catalogs or by using requisition templates that have been defined for them by purchasing managers. Requisitions can contain items from one or more suppliers. The customer server 34 automatically checks requisitions against the spending limit defined for each employee. If the requisition is within this limit, customer server automatically approves the requisition, then creates and sends an EDI formatted purchase order to each supplier included in the requisition. If the exceeds the spending limit, the customer server automatically forwards it to an authorized employee for approval.

Detailed Description Text (315):

Suppliers may receive and send the customer server documents using a Transaction Gateway. Suppliers can integrate the Transaction Gateway with their existing order processing systems.

Detailed Description Text (316):
The customer server provides support for both desktop delivery where requisitioners receive goods directly) and for warehouse and dock deliveries with a goods inward component to check goods in, raise queries about deliveries and route the goods to the appropriate requisitioner. The customer server links to the bank's existing systems to enable secure electronic payment for goods using a corporate purchasing card or using the bank's Automated Clearing House (ACH). If required, companies can also pay using their existing accounts payable systems.

Current US Original Classification (1):
705/27

Current US Cross Reference Classification (3):
705/26

CLAIMS:

1. Electronic Commerce System for procuring goods/services by a plurality of users within a custom organization, comprising

(a) a plurality of terminals;

(b) a customer server connectable to each of said terminals and including log-on means for providing access to said customer server to a user by means of one of said terminals only if the user can be properly authenticated for a predetermined level of purchasing authorization;

(c) a supplier system including a supplier catalog server for storing data representing a supplier catalog of goods/services that are available for purchase by authorized users in the customer organization, and a supplier processor server for processing orders received by the authorized user within the customer organization, said supplier system being directly accessible to said customer server through an internet connection; and

(d) security means provided within said servers which limit transactions to customers and suppliers who have pre-arranged relationship for displaying selected supplier catalog information to a user within said organization, consistent with said predetermined level of authorization, for issuing a purchase order by the user to said supplier system, and issuing an invoice by said supplier system to the customer organization after goods/services have been delivered to the user.

2. System as defined in claim 1, wherein said terminals are connectable to said customer server by means of a LAN network.

3. System as defined in claim 1, wherein said terminals are connectable to said customer server by means of an Intranet connection.

4. System as defined in claim 1, wherein said customer server includes means for defining the level of authorization for the approval of acquisition of goods/services by a user logged on to said customer server, said level of authorization also defining pre-selected goods/services that the user has available for viewing from said supplier catalog.

5. Suppler System as defined in claim 1, wherein said customer server is provided with means for linking said customer server with a purchaser legacy system.

6. Supplier System as defined in claim 1, wherein said customer server output data format and bank server input data format are compatible and said customer and bank servers are directly coupled to each other over an Internet connection.

7. Supplier System as defined in claim 1, wherein said customer server output data format and bank server input data format are incompatible, and further comprising a clearing house gateway between said customer and bank servers for translating said output data format to be compatible with said input data format.

8. System as defined in claim 1, wherein said supplier system includes a supplier legacy catalog, said merchant catalog server including means for accessing said supplier legacy catalog to make same accessible for viewing by said users in said customer organization.

9. System as defined in claim 1, wherein said supplier system includes a supply legacy system, said supplier processor server including means for accessing said supply legacy system.

10. System as defined in claim 1, further comprising counting means at a service bureau for counting the number of purchase orders issued by said customer server to said suppler system, whereby said service bureau may be compensated for hardware, software and/or services in the use of the system.

11. System as defined in claim 1, wherein said customer server, supplier system and bank server, supplier system and bank server Internet connections use at least one of the following electronic exchange protocols: EDI (ANSI and EDIFACT), OBI (open buying on the Internet), S/MIME, MIME, SMTP, HTTP, and TCP/IP.

12. Electronic Commerce System as defined in claim 1, wherein further comprising a bank server accessible by said customer server through an Internet connection for payment to said supplier for the procure goods/services upon receiving instructions to make payment by said customer server.

13. Systems as defined in claim 1, further comprising a certificate authority for uniquely authenticating customer and suppliers to each other, whereby secure connections to exchange information and documents.

14. System as defined in claim 13, wherein said certificate authority comprises means for generating public/private keys unique to each buyer and supplier to enable encryption/decryption, authentication and integrity of all communications and/or messages transmitted between said customer server and said supplier system.

15. Electronic Commerce System for procuring goods/services by a plurality of users within an organization, comprising user hierarchy-based communication system for

(a) a supplier computer system including a supplier catalog and an order processor, said supplier catalog containing information regarding all of said supplier's goods/services and a specific profile for the purchasing organization;

(b) a procurement computer system including a plurality of terminals for use by a plurality of users within a purchasing organization each assigned an organization user profile which specifies a level of authorization for approval of the acquisition of goods and/or services from at least one predetermined supplier of goods, said procurement computer including means for displaying pre-selected goods/services on a terminal available for acquisition from said at least one supplier consistent with the user's level of authorization for the acquisition of goods/services from said supplier;

(c) a communication link for selectively accessing, for downloading by a user, selected information from said supplier catalog to the user's terminal to said

procurement computer and the supplier system being programmed to establish a cryptographically secure session for ordering and filling an order of goods/services, by means of said order processor, from said supplier, only when an authorized user seeks to acquire one or more products/services which the user is authorized to purchase.

16. Method of procuring goods/services by any one of a plurality of users within a customer organization from a supplier, comprising the steps of:

(a) logging on by a user on a terminal to a customer server at the customer organization;

(b) authenticating the user as a valid user;

(c) connecting the user to a supplier catalog over the Internet and displaying selected goods/services on the user terminal consistent with the user's authorization profile established by the customer authorization;

(d) completing a requisition request by the user selecting the products/services the user wants to purchase;

(e) sending the requisition requests to the customer server and checking the request against the user's purchasing authorization limits; and

(f) creating a purchase order for the supplies only if the user's purchasing authorization limits are not exceeded.

17. Method as defined in claim 16, wherein the user is connected to the customer server by means of an Intranet connection.

18. Method as defined in claim 16, further comprising the step of sending a message to the user's designated supervisor to approve, alter or cancel the requisition request where the user's purchasing authorization limits have been exceeded.

Previous Doc Next Doc Go to Doc#

Set	Items	Description
S1	1	ACTIVEX AND (AUTHENTIC? (5W) PRICE) AND PD<=001215
S2	1	ACTIVEX AND (AUTHENTIC? (5W) PRICE)
S3	4	ACTIVEX AND (AUTHENTIC? (S) PRICE)
S4	3	RD (unique items)

?

T S4/3,KWIC/1-3

4/3,KWIC/1
DIALOG(R)File 647:CMP Computer Fulltext

01158006 CMP ACCESSION NUMBER: NWC19980401S0016
Request For Proposal:Security Services
Greg Schipley
NETWORK COMPUTING, 1998, n 906, PG52
PUBLICATION DATE: 980401
JOURNAL CODE: NWC LANGUAGE: English
RECORD TYPE: Fulltext
SECTION HEADING: Features
WORD COUNT: 6587

... evaluate the security interface and the ability of Internet/intranet scripting languages (such as CGI, ActiveX and Java) to support the integrity of transactions being transmitted and processed." Far too many...Waterhouse brings invaluable diversity to organizations requiring inspections of this nature.

In addition to depth, Price Waterhouse's solution discloses a fair amount of thought-provoking detail. For example, it explores...

...many of them know whether those threats remain after deployment of session encryption, key-based authentication and a strong telephone infrastructure?

Price Waterhouse points out that, all too frequently, higher-end...

4/3,KWIC/2
DIALOG(R)File 647:CMP Computer Fulltext

01137199 CMP ACCESSION NUMBER: CRN19970908S0098
Internet Security:Being Wired Has Its Price
COMPUTER RESELLER NEWS, 1997, n 753, PG134
PUBLICATION DATE: 970908
JOURNAL CODE: CRN LANGUAGE: English
RECORD TYPE: Fulltext
SECTION HEADING: White Paper:Microsoft Corp.
WORD COUNT: 2643

Until we introduced Authenticode last August, most people looked at browsing the Internet as a spectator sport. It didn...

...hole raises awareness that the really bad guys are right behind. Being wired has its price .

It follows that security and privacy in this new generation of the Internet has to...disallow execution.

Unfortunately, this explosion of functionality available for the Web-Java with capabilities, scripting, ActiveX -has brought with it an explosion of complexity. Previous security schemes have depended on an...

...for all Web sites and pages encountered. Either Java was always on or always off. ActiveX controls were always enabled or always disabled. This simplistic scheme is now a victim of...

...to perform applicationlike functionality (writing to the disk, network I /O) or they could use ActiveX controls to do the same thing. It would no longer be reasonable to allow all...

...then dutifully asks if Site A can:write to the disk, perform network operations, run ActiveX controls, script ActiveX controls, use more than 1 Mbyte of memory, perform cross-frame operations.

Huh? Are these...

...things:group sets of sites together and assign security settings to that zone.

Settings for ActiveX control download and installation, scripting, cookie management, password authentication, cross-frame security and, of course...

4/3,KWIC/3

DIALOG(R)File 647:CMP Computer Fulltext

01090268 CMP ACCESSION NUMBER: IWK19960506S0088

The Components Of Change - Companies turn to prewritten software rather than code from scratch

Jacques Surveyer

INFORMATIONWEEK, 1996, n 578, PGAD01
PUBLICATION DATE: 960506
JOURNAL CODE: IWK LANGUAGE: English
RECORD TYPE: Fulltext

SECTION HEADING: Application Development
WORD COUNT: 2110

... is that of smart objects such as Microsoft's VBX and OCX components-now dubbed ActiveX controls-or the IBM/Apple-backed OpenDoc components. These components link into such popular application...

...various user or GUI events with a battery of functions or methods.

The problem with ActiveX and OpenDoc is that both are undergoing constant change.

Visual programming, use of objects, and...

...such as Delphi, MicroFocus' Cobol, and Powersoft/Watcom's Optima will greatly ease application development. ActiveX and OpenDoc will be used as interoperable components that can be manipulated by 3GL tools. We'll see ActiveX components for the Macintosh and Unix environments, and OpenDoc on the various flavors of Microsoft Windows and Unix. But ActiveX and OpenDoc work across programming languages, unlike many object and class frameworks.

Component-based visual...

...browser takes care of the GUI and network interface and a lot of security and authentication issues. HTML Web pages are easy to create, modify, and use. JavaScript and Java applets...

...Browsers can provide direct access and interaction with multiple databases, including legacy data. And the price is right.

But a number of IS shops are skeptical about the role of Web...

?